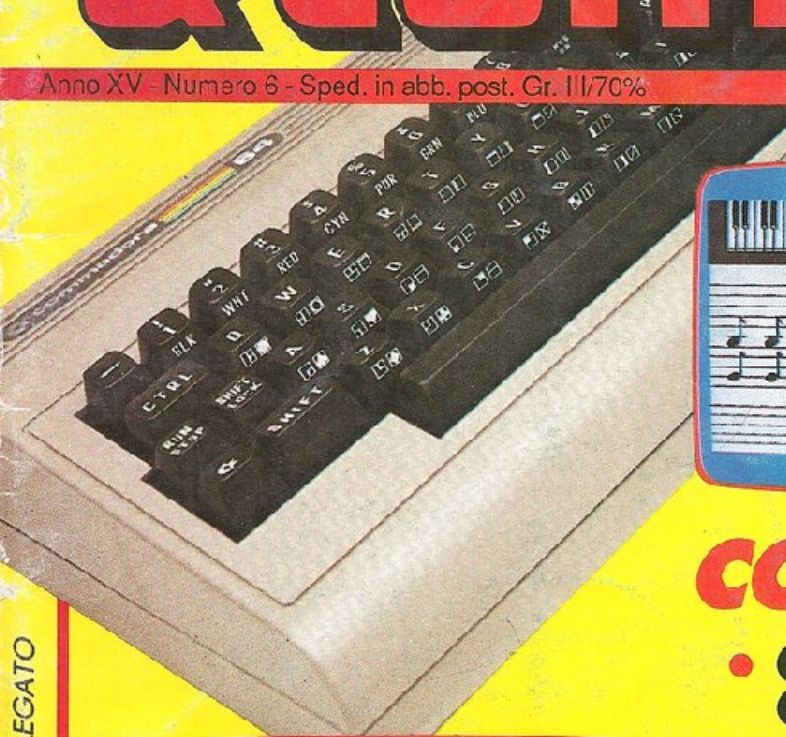


Radio Elettronica & Computer

L'UNICA RIVISTA
CON TUTTI I
PROGRAMMI
SU
CASSETTA

Anno XV - Numero 6 - Sped. in abb. post. Gr. 11/70%

GIUGNO 1986 - L. 6.000



COMMODORE 64:

- GIOCHI/PROFESSIONALI? CREALI CON GAME MAKER
- UTILITY/LE MAGNIFICHE TRE
- TV MANAGER/UN DATABASE SCEGLIE I PROGRAMMI
- WARGAME/NAPOLEONE 1802
- DEDALO/SI SALVI CHI PUÒ
- GESTIONE FILE/MEGLIO AD ACCESSO CASUALE
- LINGUAGGIO MACCHINA III



TASSA PAGATA PER CAMPIONE ALLEGATO

SPECTRUM

- FORMULA UNO/AVERE IN PUGNO IL CAMPIONATO
- JUDO/MAESTRI IN DUE PUNTATE
- SOLIDI/VIVA LA TRIGOGRAFICA!

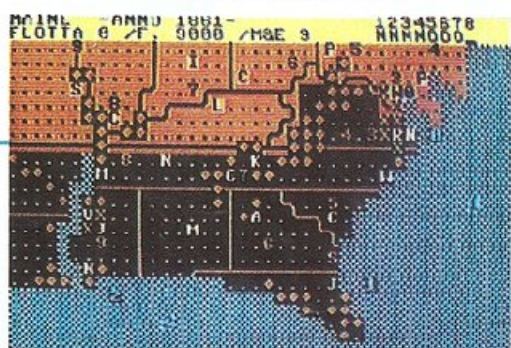


MSX: MILLE E UNA NOTA

Il mensile con disco programmi
per Commodore 64 e C128

Sped. in Abb. Postale Gr. 111770%

COMMO **in tutte** DISK **le edicole**



WARGAME

La guerra di secessione sul computer

Anno I - Numero 2 - MAGGIO 1986 - L.13.000



RICETTE

Il cucinone elettronico



FANTASY

Meandro nel paese delle meraviglie: il primo gioco d'avventura tutto in italiano



HIT-PARADE

Un archivio elettronico per tenere in ordine tutti i dischi e le cassette

DIREZIONE GENERALE E
AMMINISTRAZIONE

Editronica srl

20122 Milano - C.so Monforte 39
Ufficio abbonamenti 02/702429

Radio
**Editronica
& Computer**

Direttore Responsabile
Stefano Benvenuti

Coordinamento editoriale
Fancesca Marzotto

Impaginazione elettronica
Adelio Barcella

Collaboratori
Giorgio Caironi
Marco Gussoni
Mario Magnani
Ivonne Rossi
Guido Ricciardi

Pubbliche Relazioni
Mauro Gandini

SERVIZIO ABBONAMENTI

Editronica srl - C.so Monforte 39 - Milano
Conto Corrente Postale n. 19740208
Una copia L. 6.000 - Arretrati: il doppio
del prezzo di copertina. Abbonamento 12
numeri L. 60.000 (estero L. 80.000). Peri-
odico mensile. Stampa: "VEGA sas"
Via Teodosio 17, Milano. Distribuzione
esclusiva per l'Italia A.&C.Marco
S.p.A. - Via Forzezza 27 - 20125 Mila-
no. Tel. 02/25261 - Telex 350320. ©
Copyright 1986 by Editronica srl. Regi-
strazione Tribunale di Milano N. 112/72
del 17.3.72. Pubblicità inferiore al 70%

Tutti i diritti di riproduzione e traduzione
di testi, articoli, progetti, illustrazioni,
disegni, circuiti stampati, listati dei
programmi, fotografie ecc. sono riservati
a termini di legge. Progetti, circuiti e
programmi pubblicati su RadioELET-
TRONICA & COMPUTER possono es-
sere realizzati per scopi privati, scien-
tifici e dilettantistici, ma ne sono vietati
sfruttamenti e utilizzazioni commerciali.
La realizzazione degli schemi, dei proget-
ti e dei programmi proposti da Radio-
ELETTRONICA & COMPUTER non
comporta responsabilità alcuna da parte
della direzione della rivista e della casa
editrice, che declinano ogni responsabi-
lità anche nei confronti dei contenuti delle
inserzioni a pagamento. I manoscritti, i
disegni, le foto, anche se non pubblicati,
non si restituiscono.



RadioELETTRONICA & COMPUTER
è titolare in esclusiva per l'Italia dei testi
e dei progetti di Radio Plans e Electro-
nique Pratique, periodici del gruppo So-
ciété Parisienne d'Édition.

SOMMARIO

GIUGNO 1986 - ANNO XV - NUMERO 6

- 6** **TRE UTILITY IN LINGUAGGIO MACCHINA**
Una è utile, una è necessaria, una è indispensabile: sono tre, e sono scritte nel linguaggio più avanzato per il vostro C 64. Stoplist, Stopsystem e On Error Goto vi risparmieranno moltissime perdite di tempo.
- 10** **MSX: TUTTA LA MUSICA CHE VUOI**
Dopo Play, è ora la volta di Sound, l'istruzione che proclama definitivamente musicista il vostro MSX. C'è anche un listato (il relativo programma è nella cassetta allegata), per esercitarsi subito con la bacchetta in mano.
- 15** **TV MANAGER 64**
Tutte le sere la stessa storia: scene di guerra per il possesso del telecomando. Non sarebbe meglio pianificare prima? Decidere con metodo che cosa si guarderà alla TV? Questo database vi aiuterà a scegliere i programmi.
- 19** **CAMPIONATO DI FORMULA UNO**
Classifiche relative e assolute; piloti e team, un Gran Premio dopo l'altro, per una gestione meticolosa e completa di un intero campionato. Sullo Spectrum, naturalmente. E i dati da inserire li trovate qui!
- 23** **COMMODORE 64: FARSI I GIOCHI**
Diciamolo: i giochi fatti in casa sono al 90% da buttar via. A meno che non si sia campioni in linguaggio macchina e si abbiano sei mesi da passare al video. Oppure ci vuole Game Maker, il favoloso programma che...
- 29** **JUDO: CAMPIONI D'ORIENTE**
Si chiama judogi, il costume bianco da combattimento che si indossa, tradizionalmente, per allenarsi nel judo. La prima volta che caricherete su Spectrum il programma Judo, userete lo judogi con cintura bianca. Poi...
- 31** **WAR GAME: NAPOLEONE IL GRANDE**
Per tutti gli appassionati di giochi di simulazione bellica: guerra napoleonica, per Commodore 64, è un programma che permette di giocare tenendo conto di moltissime variabili, determinanti per le sorti del combattimento.
- 33** **DENTRO IL LABIRINTO**
Da sempre affascina la fantasia dell'uomo, il misterioso labirinto. Viviamolo dall'interno, in una simulazione prospettica ben riuscita. Undici livelli di difficoltà, un implacabile timer, ma anche un misericordioso tasto di aiuto.
- 35** **TRE DIMENSIONI PER I SOLIDI**
Creato per lo Spectrum 16, questo programma potrà sembrare un po' lento ai possessori del 48K, ma niente paura: adattarlo è facile. Con due punti di fuoco e una linea di terra, crea parallelepipedi con formule trigonometriche.
- 37** **C 64: FILE AD ACCESSO CASUALE**
File sequenziali? No, grazie. Questa utility, e le mini routine che vi consente di creare, permettono di gestire i file ad accesso casuale: relativi, gestiti dal sistema, e random, gestiti dall'utente.
- 40** **LINGUAGGIO MACCHINA PER C64**
Terza puntata del corso di linguaggio macchina, accompagnato, in questo numero, anche da tre ottime utility (pag. 6). Dedicato da RadioEditronica & Computer a tutti coloro che pretendono il massimo dal loro computer.

RUBRICHE: Novità, pagina 4 - Vorrei sapere, vorrei proporre, pagina 48.

Caricate così i programmi della cassetta allegata:

Lato C64. Riavvolgete il nastro e premete SHIFT RUN/STOP sulla tastiera del C64 e PLAY sul registratore. Verrà caricato il turbo tape che a sua volta caricherà il programma di presentazione con il menù dei programmi. Terminata la presentazione, per caricare uno qualsiasi dei programmi è sufficiente digitare: <-L 'NOME PROGRAMMA' seguito dalla pressione del tasto RETURN.
Lato Spectrum. Riavvolgete il nastro e caricate con LOAD "" seguito da ENTER, quindi selezionate l'opzione corrispondente al programma desiderato.

Chi, Cosa, Come, Quando...

Il terminale miniaturizzato Dt-6000 Casio ha una stampante incorporata e con una penna ottica è in grado di leggere i codici a barre.

Lauree milionarie

L'Ibm Italia assegnerà 25 premi del valore di sei milioni di lire ciascuno per tesi di laurea sull'informatica, sull'intelligenza artificiale, sull'ingegneria del software, sulle telecomunicazioni e sulle applicazioni gestionali e tecnico-scientifiche. Le borse di studio saranno riservate a giovani che si siano laureati con il massimo dei voti in ingegneria, economia, fisica, matematica, informatica e statistica tra il settembre 1985 e il giugno di quest'anno. Per ulteriori informazioni sulle modalità e sui termini di ammissione al concorso è possibile rivolgersi alla Direzione relazioni universitarie e scientifiche, via Giorgione 159, Roma, tel. 06/54864568.

Nuovo look per il video

I nuovi videoregistratori Family line della giapponese Hitachi si fanno notare, per prima cosa, per un'estetica aggressiva e per un design capace di ben fi-

gurare in qualsiasi arredamento; la tastiera è stata studiata in base ai più recenti dettami dell'ergonomia, mentre un telecomando a raggi infrarossi si incarica di trasmettere a distanza i comandi. Ma il segreto del successo di questi apparecchi è una nuova circuitazione, che garantisce una migliore definizione dell'immagine rispetto ai modelli precedenti: tra le funzioni disponibili si fanno notare quella denominata Auto Operate On (cioè l'accensione automatica all'atto dell'inserimento della cassetta), la memoria con preselezione di 79 canali, la funzione di Auto repeat per l'intero nastro o per una sua parte determinabile dall'utente e, infine, il sistema battezzato Irt, con il quale si può impostare, oltre alla durata, l'inizio della registrazione. I modelli disponibili sono due, siglati rispettivamente Vt 110 e Vt 120 (1,090 e 1,220 milioni di lire). Il secondo dispone di un telecomando capace di tenere sotto controllo ben 17 funzioni.

La Hitachi ha anche lanciato una nuova linea di televisori contrassegnati dalla sigla Sqf, con dimensioni di 21, 25 e 28 pollici, dotati di schermo piatto autoconvergente. Ciò significa, in concreto, che la visione resta perfetta anche quando lo spettatore non è in asse con il centro dell'apparecchio. Il prezzo dei televisori della linea Sqf non è stato ancora fissato.

Terminale da viaggio

Si chiama Dt-6000, ha una stampante incorporata e dispone di una memoria Ram espandibile fino a 96K: è il nuovo terminale portatile della Casio che con due tastiere distinte (la prima è numerica, con 10 tasti di funzione programmabili, mentre la seconda è alfanumerica, anch'essa programmabile in C 63 Basic residente) consente di trasmettere e ricevere dati nelle più svariate situazioni. A confermare la flessibilità d'u-



so di questo terminale miniaturizzato è la presenza delle interfacce incorporate Rs-232, interamente programmabile, Bcr (per penna ottica), che è in grado di leggere codici a barre, e Mcr, per la lettura di schede magnetiche. Dt-6000 è, insomma, capace di rendersi utile in tutte le situazioni che richiedono velocità e sicurezza nella raccolta dei dati, dalla lettura dei contatori alla gestione degli ordini, dall'elaborazione di calcoli topografici alla proiezione di dati economico-finanziari, con un solo limite: la fantasia di chi utilizza questo piccolo prodigio dell'elettronica. Il Dt-6000 costa 1,650 milioni di lire (Iva esclusa).

Computer Play 1986

Quest'anno la manifestazione Computer Play si svolgerà dal 4 all'8 settembre, nell'ambito del Sim Hi-fi-Ives (Salone Internazionale della Musica e High Fidelity e International Video and Consumer Electronics Show), a Milano, e presenterà un nutrito pacchetto di incontri e seminari di grandissimo interesse per tutti coloro che si occupano di programmazione di giochi o che vogliono avvicinarsi per la prima volta alle problematiche del gioco elettronico.

La partecipazione ai corsi e ai seminari (il programma può essere richiesto scrivendo alla segreteria generale della manifestazione, a Milano, via Dome-

I nuovi videoregistratori Hitachi Family line dispongono di una memoria per la preselezione di 79 canali e di un telecomando che tiene sotto controllo 17 funzioni.



nichino 11, tel. 02/4815541) non richiede requisiti specifici. Al fine di limitare la partecipazione soltanto alle persone veramente motivate, invece, verrà richiesta una tassa di iscrizione simbolica di 10 mila lire per ciascuna giornata (tranne che per quella conclusiva), pagabile al momento dell'ingresso nella sala conferenze: ciascuno sarà poi libero di seguire uno solo o più seminari, a seconda dei suoi interessi specifici. Gli argomenti degli incontri saranno trattati dai maggiori esperti italiani dei singoli settori e verranno illustrati dal vivo, con l'aiuto di calcolatori.

Software e diritto d'autore

Da pochi giorni è disponibile al pubblico il volume *Software e diritto d'autore*, che raccoglie gli atti del convegno organizzato recentemente su questo tema dal Ceiiil (Centro europeo informazione informatica e lavoro), in collaborazione con la Ibm Italia.

Nel volume si rileva come nella maggior parte dei paesi più avanzati siano già state adottate concrete misure di tutela del software, dal brevetto all'applicazione di provvedimenti contro la concorrenza sleale. In Italia, invece, le leggi o sono carenti o non vengono applicate. Eppure, sostengono i personaggi che con i loro interventi hanno contribuito alla stesura del volume, c'è un ampio accordo tra giuristi, operatori del settore e utenti su di un punto: l'attuale legge, nata per tutelare le opere di ingegno, sarebbe già adeguata a proteggere diversi tipi di software.

Il volume *Software e diritto d'autore* (208 pagine, 18 mila lire), curato da Nadir Tedeschi e Giampio Bracchi e pubblicato dall'editore Franco Angeli, è completato da una rassegna sulla normativa internazionale e da una bibliografia che comprende i principali contributi italiani e stranieri in materia di tutela giuridica del software.

Vacanze con il computer

L'Eurolanguage, come ogni anno, propone un ampio programma di vacanze di studio per l'apprendimento delle lingue inglese, francese, tedesca e spagnola nei paesi d'origine. La qualità dei corsi è garantita dal metodo di studio attivo-diretto, con l'impiego di insegnanti madrelingua che operano con piccoli gruppi di studio; il confort degli allievi, invece, è assicurato dalla sistemazione in college e residence o presso famiglie selezionate.

Ma una delle opportunità più interessanti offerte dall'Eurolanguage sono i corsi di studio supportati dall'uso di personal computer: si terranno a Oxford, sia con residenza presso famiglie sia presso un college (quello di Westminster) e le macchine utilizzate saranno quelle della Bbc. Per essere ammessi a questi corsi all'insegna dell'informatica personale è necessario avere compiuto i 16 anni: per ulteriori informazioni è possibile rivolgersi alla Eurolanguage Italia, piazza San Sepolcro 2, Milano, telefono 02/866094.

Tempo di messaggi

Gli orologi da muro e da scrivania messi a punto dai tecnici della Levi, un'azienda milanese specializzata nella produzione di apparecchiature elettroniche miniaturizzate, hanno una caratteristica sorprendente: oltre a segnare lo scorrere del tempo, infatti, sono in grado di registrare e ripetere messaggi. In ufficio, per esempio, consentono di lasciare comunicazionini ai collaboratori senza essere costretti ad affidarsi



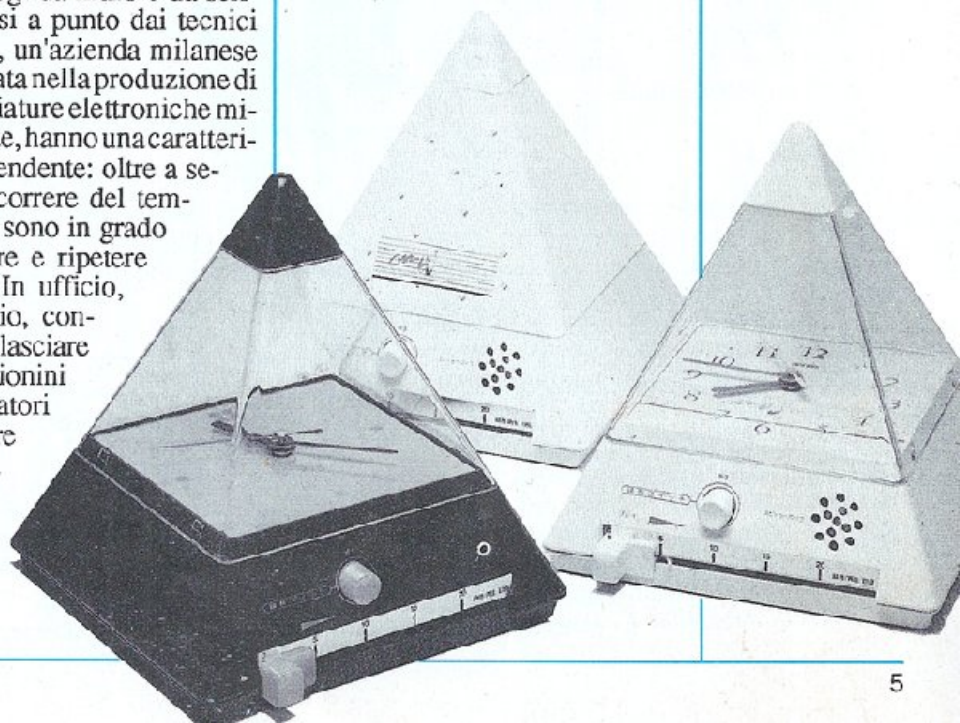
a biglietti leggibili da chiunque, e, soprattutto, spesso destinati ad andare smarriti. A casa si possono lasciare istruzioni, magari un numero telefonico dove essere agevolmente rintracciati o gli auguri per qualche ricorrenza familiare.

Questi insoliti orologi sono prodotti in un'elegante versione a muro e in una da tavolo dall'originale forma di piramide: in entrambi i casi la registrazione ha una durata di 20 secondi, mentre le batterie durano circa un anno.

La Levi ha anche presentato una sveglia del tutto particolare, che permette di vedere l'ora proiettata sul soffitto restando coricati. Questa insolita sveglia funziona sia a rete sia a batteria. Gli orologi e la sveglia sono in vendita nei negozi Ricordi.

Il tour operator milanese Eurolanguage propone una serie di vacanze di studio in Inghilterra. I corsi prevedono l'uso di personal computer.

I nuovi orologi da scrivania della Levi hanno un design originale e sono in grado di memorizzare e trasmettere messaggi.



Commodore 64



UTILITY

Tre routine in linguaggio macchina, tre validi strumenti di lavoro per chi vuole conoscere più da vicino il suo computer: Stolist blocca o rallenta il LIST di un programma, Stopssystem ferma l'esecuzione di un programma senza che sia poi necessario farlo ripartire da capo e On Error Goto evita che un programma interrompa l'elaborazione in corso perché incontra un errore.

Tre assi nella manica

Le routine proposte, Stolist, Stopssystem e On Error Goto, hanno lo scopo di facilitare alcune delle più importanti operazioni che intervengono durante la fase di stesura e correzione di un programma. On Error Goto in particolare consente di potenziare il Basic standard del Commodore 64 offrendo la possibilità di intercettare gli eventuali errori che occorrono durante la fase di esecuzione di un programma e quindi di agire in conseguenza evitando l'arresto del programma stesso. Tutte le routine sono scritte in linguaggio macchina e possono essere usate indipendentemente, quindi è possibile caricare in memoria solo quelle che si vogliono usare. Per utilizzare le routine basta digitare ed eseguire i caricatori Basic relativi (figura 1). I tre caricatori devono essere registrati su disco o nastro prima di essere eseguiti, poiché si autodistruggono dopo aver caricato in memoria i codici in linguaggio macchina.

Stolist

La prima routine permette di effettuare una pausa o di rallentare notevolmente il LIST di un programma sul video. Normalmente per ottenere una pausa durante l'esecuzione del LIST si deve premere il tasto RUN/STOP con l'inconveniente di dover ogni volta digitare nuovamente il comando LIST. Con Stolist premendo il tasto Commodore si ottiene l'interruzione del LIST del programma sino a quando non si preme un tasto qualsiasi, nel qual caso il LIST riprende dal punto in cui era stato interrotto.

Tenendo premuto il tasto Commo-

Figura 2

00001	0000			*=\$C000	
00002	C000	48		PHA	/SALVA I REGISTRI
00003	C001	98		TYA	/UTILIZZATI DALLA
00004	C002	48		PHA	/ROUTINE DI LIST
00005	C003	20	87 EA	JSR	52039
00006	C000	AD	8D 02	LDA	553
00007	C008	C9	02	CMP	#2
00008	C00E	F0	0A	BEQ	LEGGI
00009	C00D	A9	09	LDA	#0
00010	C00F	85	C8	STA	190
00011	C011	68		PLA	
00012	C012	A8		TAY	
00013	C013	68		PLA	
00014	C014	4C	1A A7	JMP	42778
00015	C017				/SALTA ALLA ROU-
00016	C017	20	87 EA	LEGGI JSR	60039
00017	C01A	A5	0B	LDA	203
00018	C01C	C9	40	CMP	#64
00019	C01E	F0	F7	BEQ	LEGGI
00020	C020	A9	00	LDA	#0
00021	C022	85	C6	STA	198
00022	C024	58		PLA	
00023	C025	A8		TAY	
00024	C026	58		PLA	
00025	C027	4C	1A A7	JMP	42778
00026	C02A			.END	

dore e digitando ripetutamente un tasto qualsiasi si può ottenere un LIST estremamente più lento di quanto si possa fare premendo il tasto CTRL. La routine occupa 41 byte a partire dalla locazione 49152 (comunque è completamente rilocabile) e si inserisce direttamente nella routine di LIST modificando il vettore di salto alla routine (locazioni 774 e 775) in modo tale che prima di procedere con la normale esecuzione del comando si passi attraverso la nuova routine. Per questo motivo, nel caso che la routine venga rilocata, è indispensabile correggere il caricatore Basic nelle linee che

modificano il vettore alla routine di LIST per fare in modo che punti alla nuova routine (cioè si devono calcolare il byte alto e basso dell'indirizzo da cui inizia la routine e memorizzarli rispettivamente nelle locazioni 774 e 775). Il listato in figura 2 è ampiamente commentato e quindi dovrebbe risultare abbastanza semplice capire il funzionamento del programma.

Stopssystem

Questa routine permette di congelare il sistema a tempo indeterminato e quindi di bloccare anche l'esecuzione



Figura 1

```

2 REM-----
4 REM LOADER/STOPLIST
6 REM-----
8 :
10 CK=0:PORT=49152 TO 49193:READ D:CK=CK+D:POKET,D:NEXT
20 IF CK<>5277 THEN PRINT"ERRORE NEI DATA":END
30 POKE774,0:POKE775,192:NEW
40 DATA 72,152,72,32,135,234,173,141
50 DATA 2,201,2,240,10,169,0,133,198
60 DATA 104,168,104,76,26,167,32,135
70 DATA 234,165,203,201,64,240,247
80 DATA 169,2,133,198,104,168,104,76
90 DATA 26,167

```

```

2 REM -----
4 REM LOADER STOPSYSTEM
6 REM -----
8 :
10 CK=0:PORT=49194 TO 49234:READ D:CK=CK+D:POKET,D:NEXT
20 IF CK<>5001 THEN PRINT"ERRORE NEI DATA":END
30 SYS49194:NEW
40 DATA 120,169,55,141,20,3,169,192
50 DATA 141,21,3,96,96,32,135,234,165
60 DATA 203,201,4,240,3,76,49,234,32
70 DATA 105,234,165,203,201,3,208,247
90 DATA 169,0,133,133,76,40,204

```

```

2 REM-----
4 REM LOADER/ON ERROR GOTO
6 REM-----
8 :
10 CK=0:PORT=49235 TO 49311:READ D:CK=CK+D:POKET,D:NEXT
20 IF CK<>8113 THEN PRINT"ERRORE NEI DATA":END
30 NEW
40 DATA 32,253,174,32,107,163,165,20
50 DATA 164,21,141,167,2,140,168,2
60 DATA 169,121,160,192,141,0,3,140
70 DATA 1,3,96,160,130,160,227,141,0
80 DATA 3,140,1,3,96,163,48,32,165
100 DATA 164,58,141,169,2,140,170,2
110 DATA 173,167,2,172,168,2,133,20
100 DATA 132,21,32,163,168,96,76,113,194,76

```

Tavola 1

Codice	Messaggio
1	too many files
2	file open
3	file not open
4	file not found
5	device not present
6	not input file
7	not output file
8	missing file name
9	illegal device number
10	next without for
11	syntax
12	return without gosub
13	out of data
14	illegal quantity
15	overflow
16	out of memory
17	undef'd statement
18	bad subscript
19	redim'd array
20	division by zero
21	illegal direct
22	type mismatch
23	string too long
24	file data
25	formula too complex
26	can't continue
27	undef'd function
28	verify
29	load

di un programma in un istante qualsiasi. Quando la routine viene disattivata il computer riprende la sua normale procedura e, se era stata interrotta l'esecuzione di un programma, questa riprende dal punto in cui si era fermata.

Questa routine si rivela molto utile quando si vuole osservare attentamente l'esecuzione di un programma per annotare le eventuali modifiche da fare.

Con Stopsystem è possibile fermare a più riprese l'esecuzione di un programma senza poi essere costretti a farlo ripartire dall'inizio e senza il pericolo che i messaggi del computer al-

terino le schermate prodotte dal programma. Infatti Stopsystem entra in funzione senza visualizzare nessun messaggio.

Per attivare la routine basta premere il tasto F1 mentre premendo F7 viene disattivata.

Questa routine non è rilocabile ma la si può facilmente rendere tale con alcune semplici modifiche: basta cambiare i valori che seguono l'istruzione LDA (figura 3) alle locazioni SC02B e SC030.

Queste due istruzioni provvedono a modificare il vettore alla routine di Interruzione (IRQ) per fare in modo che,

Figura 3

```

00001 0000          *=$C02A
00002 C02A 78      SEI          ;DISABILITA INTERRUPT
00003 C02B A9 37   LDA #<INIT    ;MODIFICA IL VETTORE
00004 C02D 3D 14 03 STA $314      ;DI INTERRUPT PER FARLO
00005 C030 A9 C0   LDA #>INIT    ;PUNTARE ALLA NUOVA ROU-
00006 C032 8D 15 03 STA $315      ;TINE
00007 C035 58      CLI          ;RIABILITA INTERRUPT
00008 C036 60      RTS          ;RITORNA AL BASIC
00009 C037 2D 87 EA INIT JSR #EA87 ;SCANSIONE TASTIERA
00010 C03A A5 CB   LDA $CB      ;SE TASTO F1 (#4) SALTA
00011 C03C C9 04   CMP #4        ;A CICLO. ALTRIMENTI ESE-
00012 C03E F0 03   BEQ CICLO     ;GUI LA NORMALE ROUTINE
00013 C040 4C 31 EA JMP #EA31    ;DI INTERRUPT
00014 C043 2D 87 EA CICLO JSR #EA87 ;SCANSIONE TASTIERA
00015 C046 A5 CB   LDA $CB      ;SE (<) TASTO F7 (#4)
00016 C048 C9 03   CMP #3        ;SALTA A CICLO. ALTRIMEN-
00017 C04A D0 F7   BNE CICLO     ;TI VUOTA IL BUFFER DI TA-
00018 C04C A9 00   LDA #0        ;STIERA E ESEGUI LA NORMA-
00019 C04E 85 C6   STA 136      ;LE ROUTINE DI INTERRUPT
00020 C050 4C 31 EA JMP #EA31
00021 C053          .END

```

Figura 4

```

00001 0000          *=$C053
00002 C053 2D FD AE JSR #AEFD    ;VERIFICA LA PRESEN-
00003 C056          ;ZA DELLA VIRGOLA
00004 C056          ;DOPO SYS49235
00005 C056 2D 6B A9 JSR #A96B    ;CONVERTE IL NUMERO
00006 C059          ;DECIMALE DOPO LA
00007 C059          ;VIRGOLA IN UN INTE-
00008 C059          ;RO A DUE BYTE
00009 C05B A5 14   LDA $14      ;LEGGE I DUE BYTE
00010 C05B A4 15   LDY $15      ;DELLA CONVERSIONE E
00011 C05D 8D A7 02 STA $02A7    ;LI PONE NELLE LOCA-
00012 C060 8C A8 02 STY $02A8    ;ZIONI $02A7 E $02A8
00013 C063 A9 79   LDA #<INIT    ;MODIFICA IL VETTO-
00014 C065 A0 C0   LDY #>INIT    ;RE ALLA ROUTINE DI
00015 C067 8D 00 03 STA $0300    ;GESTIONE DEGLI ER-
00016 C06A 8C 01 03 STY $0301    ;RORI PER FARLO PUN-
00017 C06D          ;TARE ALLA NUOVA
00018 C06D          ;ROUTINE
00019 C06D 60      RTS          ;RITORNA AL BASIC
00020 C06E A9 8E   NRM LDA #$8B    ;RIPRISTINA LA NORMA-
00021 C070 A0 E3   LDY #$E3    ;ROUTINE DI GESTIONE
00022 C072 8D 00 03 STA $0300    ;DEGLI ERRORI
00023 C075 8C 01 03 STY $0301
00024 C078 60      RTS          ;RITORNA AL BASIC
00025 C079          ;

```


prima che vengano eseguite alcune routine del sistema operativo, il computer esegua la nostra. La modifica di questo vettore può essere fatta solo tramite il linguaggio macchina e quindi

Figura 5

```

100 PRINT"3"
105 SYS49235,900
110 FORX=-10TO10
120 Y=((X+1)↑2)/X
130 PRINTX,Y
140 NEXT
145 SYS49262
150 PRINT"PREMI SHIFT"
160 WAIT653,1
165 SYS49235,950
170 FORX=5TO-1STEP-.5
180 Y=LOG(X)
190 PRINTX,Y
200 NEXT
210 PRINT"FINE"
220 SYS49262:END
900 PRINTX,"#";
    PEEK(2),"01"
910 GOTO140
950 PRINTX,"#";
    PEEK(2),"02"
960 GOTO200

```

chi non possiede un assembler per compilarla deve necessariamente modificare il caricatore Basic, poiché cercando di utilizzare in modo diretto o da programma l'usuale comando POKE si causerebbe il blocco del sistema.

On Error Goto

Fra le tre routine questa (figura 4) è forse la più utile. Mentre si può fare a meno delle comodità offerte da Stolist e Stopsystem, è impensabile riuscire a risolvere un certo tipo di situazioni senza l'ausilio di On Error Goto.

Con essa si può evitare che un programma si fermi perché è stato incontrato un errore, e non solo: si può fare in modo che il programma agisca di conseguenza.

Ad esempio in un programma che effettua lo studio di funzioni è molto utile impedire che l'elaborazione si fermi se durante i calcoli sono state eseguite operazioni illecite (per esempio la divisione per zero oppure il calcolo di una radice di ordine pari di un numero negativo).

L'unica limitazione della routine consiste nel fatto che non è in grado

di riconoscere gli errori del DOS (figura 5) e che non può essere usata in modo diretto.

Vediamo in che modo può essere usata la routine. Con SYS 49235,NL viene attivata: da questo momento ogni volta che nel corso dell'elaborazione si incontrerà un errore il programma automaticamente eseguirà un salto (un GOTO) alla linea NL (NL può essere un intero e non una espressione o una variabile) e metterà nella locazione 2 il codice dell'errore e nelle locazioni 681 e 682 il numero di linea in cui questo è avvenuto. SYS 49262 ripristina la normale routine di gestione degli errori.

E' conveniente inserire sempre questa istruzione nel programma subito dopo la parte che tratta l'errore. Questo per impedire che eventuali errori non previsti vengano ripetutamente ignorati.

La routine non controlla che il numero di linea specificato esista effettivamente. Comunque in questo caso per riprendere il controllo del sistema basta premere RUN/STP e RESTORE. Il listato della figura 5 mostra un tipico esempio di utilizzo di questa routine.

Paolo Gussoni

```

00026 C079 ;QUI INIZIA LA NUOVA ROUTINE
00027 C079 ;
00028 C079 8A INIT TXA ;SE IL CODICE DELL'ERRORE
00029 C07A 30 20 BMI NOER ;E' NEGATIVO ALLORA
00030 C07C ;NON SI TRATTA DI UN
00031 C07C ;ERRORE
00032 C07C A5 3A LDA #3A ;SE L'ERRORE VIENE
00033 C07E C8 FF CMP #FF ;FATTO IN MODO DIRET-
00034 C080 F0 1D BEQ MDIR ;TO ESEGUI LA NORMALE
00035 C082 ;ROUTINE DI ERRORE
00036 C082 86 02 STX #02 ;MEMORIZZA IN #02 IL
00037 C084 A5 38 LDA #38 ;CODICE DELL'ERRORE E
00038 C086 A4 3A LDY #3A ;IN #02A8 E #02AA LA
00039 C088 6D A8 02 STA #02A8 ;LINEA IN CUI E' AV-
00040 C08B 6C A8 02 STY #02AA
00041 C08E AD A7 02 LDA #02A7 ;PONE IN #14 E #15
00042 C091 AC A8 02 LDY #02A8 ;IL NUMERO DI LINEA
00043 C094 85 14 STA #14 ;SELEZIONATO
00044 C096 84 15 STY #15
00045 C098 20 A3 A8 EXEC JSR #ABA3 ;EFFETTUA IL SALTO
00046 C09B 60 RTS ;RITORNA AL BASIC
00047 C09C 4C 74 A4 NOER JMP #A474 ;READY PER IL BASIC
00048 C09F 4C 3A A4 MDIR JMP #A43A ;ESEQUE LA ROUTINE
00049 C0A2 ;DI ERRORE SE SI E'
00050 C0A2 ;IN MODO DIRETTO
00051 C0A2 .END

```


MSX MUSICA

Dopo PLAY, in questa ultima puntata sullo standard MSX viene presentato il comando SOUND, che permette effetti musicali particolari. E, per finire in bellezza, c'è anche un listato.



Vai col Sound!

SOUND è una particolare istruzione che permette di caricare dei valori direttamente nei registri del generatore di suono programmabile, e consente in questo modo di ottenere particolari effetti musicali altrimenti non generabili dal pur efficace comando PLAY appena visto.


I registri del generatore sono 16, dai quali 14 programmabili dal coman-


do SOUND, specificando al seguito la relativa variabile, compresa tra 0 e 13, in base alla **tavola 1**.

Oltre alla variabile che indica il numero di registro scelto, deve esserne specificata una seconda (compresa tra 0 e un massimo di 255) che indica il dato inserito nel registro stesso, e che va determinata in base a formule matematiche e particolari trasformazioni

numeriche piuttosto complicate, la cui esposizione richiederebbe lunghe divagazioni teoriche e molti esempi pratici; per questo motivo pubblichiamo Sound Coder, un utile e potente programma che, se fatto girare sul computer MSX, è in grado di fornire direttamente i valori da inserire nei registri specificando semplicemente i dati finali desiderati per l'output del suono.

Sound Coder si compone di un solo file (listato 1) che può essere salvato su cassetta con CSAVE "SND.CD", 1, caricato con CLOAD"" (in un tempo di 53 secondi) e fatto girare tramite RUN. E' composto da 152 linee Basic numerate da 10 a 765; la 10 e la 60 comprendono caratteri grafici ottenibili premendo:

GRAPH+SHIFT+I, cioè 

e GRAPH+I, cioè 

Analogamente, alle linee 80, 420 e 425 è presente il carattere grafico:



da inserire digitando GRAPH+SHIFT+X, così come nelle linee dalla 680 alla 695 si trovano GRAPH+SHIFT+;, GRAPH+;, GRAPH+U, GRAPH+2 e GRAPH+SHIFT+O, che sono rispettivamente:



TAVOLA 1

Numero di registro	Funzione	Valore massimo inseribile
0	frequenza del canale A	255
1		15
2	frequenza del canale B	255
3		15
4	frequenza del canale C	255
5		15
6	frequenza del rumore	31
7	selezione del canale	63
8	volume del canale A	16
9	volume del canale B	16
10	volume del canale C	16
11	modulazione	255
12		255
13	forma d'onda	14
14	non programmabile	-
15	non programmabile	-

Listato 1: SOUND CODER

```

10 / _____
15 / * SOUND CODER *
20 / LISTATO 1
25 / FILE: SND.CD
30 / master
35 / c 1986 by DISCOVOGUE
40 /
45 / DISCOVOGUE
50 / P.O. BOX 495
55 / 41100 MODENA ITALY
60 / _____
65 SCREEN0:COLOR14,1:KEYOFF:CLS
70 RESTORE0
75 FORN=1:TO17:READY:READS#:GOSUB420:L
OCATED,Y,0:PRINTS#:NEXT
80 DATA1,"1",2,"2",3,"* S O U N D
CODER *",4,"2",5,"* c 1
986 by DISCOVOGUE *",6,"2",7,"1":
FQ=1996750#
85 DATA10," 1 ..... FREQUENZA CAN
ALE A"
90 DATA11," 2 ..... FREQUENZA CAN
ALE B"
95 DATA12," 3 ..... FREQUENZA CAN
ALE C"
100 DATA13," 4 ..... FREQUENZA
RUMORE"
105 DATA14," 5 ..... SELEZIONE
CANALE"
110 DATA15," 6 ..... VOLUME CA
NALE A"
115 DATA16," 7 ..... VOLUME CA
NALE B"
120 DATA17," 8 ..... VOLUME CA
NALE C"
125 DATA18," 9 ..... MODLL
AZIONE"
130 DATA19," 0 ..... FORMA
D'ONDA"
135 IK=12:PA=14:K#=INKEY#:IFK#="0"THE
NIN=0:GOTO400
140 IFK#="1"THENIN=1:GOTO190
145 IFK#="2"THENIN=2:GOTO215
150 IFK#="3"THENIN=3:GOTO240
155 IFK#="4"THENIN=4:GOTO265
160 IFK#="5"THENIN=5:GOTO285
165 IFK#="6"THENIN=6:GOTO315
170 IFK#="7"THENIN=7:GOTO335
175 IFK#="8"THENIN=8:GOTO355
180 IFK#="9"THENIN=9:GOTO375
185 BEEP:GOTO135
190 S#="FREQUENZA CANALE A":GOSUB470
195 R1=0:R2=1:LI=31:LS=249594!:S#="IN
SERIRE IL VALORE (HERTZ)":GOSUB535
200 HE=FQ/16/RI:HZ=INT(HE):IFHE-HZ>.5
THENHZ=HZ+1
205 H2=INT(HZ/256):H1=HZ-H2*256
210 GOSUB590:GOTO480
215 S#="FREQUENZA CANALE B":GOSUB470
220 R1=2:R2=3:LI=31:LS=249594!:S#="IN
SERIRE IL VALORE (HERTZ)":GOSUB535
225 HE=FQ/16/RI:HZ=INT(HE):IFHE-HZ>.5
THENHZ=HZ+1
230 H2=INT(HZ/256):H1=HZ-H2*256
235 GOSUB590:GOTO480
240 S#="FREQUENZA CANALE C":GOSUB470
245 R1=4:R2=5:LI=31:LS=249594!:S#="IN
SERIRE IL VALORE (HERTZ)":GOSUB535
250 HE=FQ/16/RI:HZ=INT(HE):IFHE-HZ>.5
THENHZ=HZ+1
255 H2=INT(HZ/256):H1=HZ-H2*256
260 GOSUB590:GOTO480
265 S#="FREQUENZA RUMORE":GOSUB470
270 R1=6:LI=3962:LS=249594!:S#="INSER
IRE IL VALORE (HERTZ)":GOSUB535
275 HE=FQ/16/RI:H1=INT(HE):IFHE-H1>.5
THENH1=H1+1
280 GOSUB575:GOTO480
285 S#="SELEZIONE CANALE":GOSUB470
290 R1=7:H1=63:Y=11:S#="SPECIFICARE I
L MIX DESIDERATO":GOSUB605:Y=12:S#="P
ER OGNI CANALE":GOSUB605
295 S#="A":SR=3:SS=0:GOSUB615
300 S#="B":SR=4:SS=1:GOSUB615
305 S#="C":SR=5:SS=2:GOSUB615
310 GOSUB575:GOTO480
315 S#="VOLUME CANALE A":GOSUB470
320 R1=8:LI=0:LS=16:S#="INSERIRE IL V
ALORE":GOSUB535
325 HE=RI:H1=INT(HE):IFHE-H1>.5THENH1
=H1+1
330 GOSUB575:GOTO480
335 S#="VOLUME CANALE B":GOSUB470
340 R1=9:LI=0:LS=16:S#="INSERIRE IL V
ALORE":GOSUB535
345 HE=RI:H1=INT(HE):IFHE-H1>.5THENH1
=H1+1
350 GOSUB575:GOTO480
355 S#="VOLUME CANALE C":GOSUB470
360 R1=10:LI=0:LS=16:S#="INSERIRE IL
VALORE":GOSUB535
365 HE=RI:H1=INT(HE):IFHE-H1>.5THENH1
=H1+1
370 GOSUB575:GOTO480
375 S#="MODULAZIONE":GOSUB470

```

(segue)


```

380 R1=11;R2=12;LI=.11902;LS=15600;S#
="INSERIRE IL VALORE (HERTZ)":GOSUB53
5
385 HE=FQ/256/R1;HZ=INT(HE):IFHE-HZ.<
5THENHZ=HZ+1
390 H2=INT(HZ/256);H1=HZ-H2*256
395 GOSUB590:GOTO480
400 S#="FORMA D'ONDA":GOSUB470
405 R1=13:H1=0
410 GOSUB680
415 GOSUB575:GOTO480
420 IFS#="1"THENS#=".....
....."
425 IFS#="2"THENS#="
"
430 IFS#="3"THENS#="
"
435 IFS#="4"THENS#="
"
440 RETURN
445 A#="S10T255L6405CDEF"
450 B#="S10T255L3208ABCD"
455 PLAYA#,B#
460 FORN=8TO21:LOCATE0,N:PRINT"
"
465 NEXT:RETURN
470 GOSUB445:COLORIK,PA:LOCATE19-LEN(
S#)/2,9:PRINTS#:GOSUB515:FORG=1TO8:GO
SUB525:NEXT:GOSUB515
475 RETURN
480 LOCATE6,21:PRINT"PER ESEGUIRE ALT
RI CALCOLI":LOCATE11,22:PRINT"PREMERE
UN TASTO"
485 K#=INKEY#:IFK#<>" "THENGOTO500
490 BEEP
495 GOTO480
500 A#="S8T255L6408CFED"
505 B#="S8T255L3206ABCD"
510 PLAYA#,B#:GOTO65
515 S#="1":GOSUB420:LOCATE3:PRINTS#
520 RETURN
525 S#="2":GOSUB420:LOCATE3:PRINTS#
530 RETURN
535 LOCATE19-LEN(S#)/2,12:PRINTS#
540 S#="[MINIMO"+STR$(LI)+", MASSIMO"
+STR$(LS)+"]"
545 LOCATE19-LEN(S#)/2,13:PRINTS#
550 LOCATE3,21:INPUTR#:RI=VAL(R#):IFR
I<LIDORRI>LSTHENBEEP:S#="3":GOSUB420:L
OCATE3,21:PRINTS#:GOTO550
555 S#="3":GOSUB420:LOCATE3,21:PRINTS
#
560 S#="VALORE INSERITO:"+STR$(R1)
565 LOCATE19-LEN(S#)/2,15:PRINTS#

```

```

570 RETURN
575 S#="SOUND"+STR$(R1)+", "+STR$(H1)
580 LOCATE19-LEN(S#)/2,17:PRINTS#
585 RETURN
590 S#="SOUND"+STR$(R1)+", "+STR$(H1)+
" SOUND"+STR$(R2)+", "+STR$(H2)
595 LOCATE19-LEN(S#)/2,17:PRINTS#
600 RETURN
605 LOCATE19-LEN(S#)/2,Y:PRINTS#
610 RETURN
615 Y=14:S#=">>>>> CANALE "+S#+" <<<<
<":GOSUB605
620 Y=15:S#="1 ..... NIENETE":GO
SUB605
625 Y=16:S#="2 ..... SOLO SUONO":GO
SUB605
630 Y=17:S#="3 ..... SOLO RUMORE":GO
SUB605
635 Y=18:S#="4 ... SUONO E RUMORE":GO
SUB605
640 K#=INKEY#:IN=VAL(K#):IFIN=1THENG0
TO665
645 IFIN=2THENH1=H1-2^SS:GOTO665
650 IFIN=3THENH1=H1-2^SR:GOTO665
655 IFIN=4THENH1=H1-2^SS-2^SR:GOTO665
660 BEEP:GOTO640
665 FORY=14TO18:S#="4":GOSUB420:GOSUB
605:NEXT
670 PLAY"S8T255L6406C"
675 RETURN
680 Y=12:S#="1> | \_____ 5> | \ / / / /
^":GOSUB760
685 Y=14:S#="2> / | \_____ 6> / \ / / /
^":GOSUB760
690 Y=16:S#="3> / \ / \ / \ / \ 7> | \ / \ / \
| \":GOSUB760
695 Y=18:S#="4> | \ | \_____ 8> / | / | / |
/ |":GOSUB760
700 K#=INKEY#:IN=VAL(K#):IFIN=8THENH1
=12:GOTO745
705 IFIN=1THENH1=1:GOTO745
710 IFIN=2THENH1=4:GOTO745
715 IFIN=3THENH1=13:GOTO745
720 IFIN=4THENH1=11:GOTO745
725 IFIN=5THENH1=10:GOTO745
730 IFIN=6THENH1=14:GOTO745
735 IFIN=7THENH1=8:GOTO745
740 BEEP:GOTO700
745 FORY=12TO18:S#="4":GOSUB420:GOSUB
605:NEXT
750 PLAY"S8T255L6406C"
755 RETURN
760 LOCATE6,Y:PRINTS#
765 RETURN

```

(fine)

TAVOLA 2

10 SOUND 0,156: SOUND 1,0	frequenza del canale A a 800 Hz
20 SOUND 2,62: SOUND 3,0	frequenza del canale B a 2000 Hz
30 SOUND 4,160: SOUND 5,1	frequenza del canale C a 300 Hz
40 SOUND 6,15	frequenza del rumore a 8320 Hz
50 SOUND 7,24	solo suono sui canali A e B più suono e rumore sul canale C
60 SOUND 8,18	volume sul canale A a 12
70 SOUND 9,14	volume sul canale B a 14
80 SOUND 10,15	volume sul canale C a 15
90 SOUND 11,12: SOUND 12,3	ciclo di modulazione a 10 Hz
100 SOUND 13,4	forma d'onda di tipo 4

Il Menù principale (a sfondo nero) chiede di scegliere, tramite i tasti numerici da 0 a 9, una delle dieci opzioni di calcolo disponibili (corrispondenti alle altrettante funzioni previste dalla tavola 1): non appena un tasto viene premuto lo schermo diventa bianco e mostra, con caratteri verdi, la videata relativa all'opzione selezionata.

Come funziona

Il programma è già stato protetto contro ogni errore di immissione dei dati richiesti (limiti inferiore e superiore, correzioni, conversione e formattazione degli input), ma questo non toglie che debba sempre essere riservata massima correttezza (formale e sostanziale) soprattutto in questa fase operativa.

Le 10 opzioni selezionabili sono raggruppabili in 6 gruppi:

1) Determinazione delle frequenze dei canali A, B e C (opzioni 1, 2 e 3): si deve inserire il valore della frequenza desiderata in output (espressa in hertz), che deve essere compreso tra 31 e 249.594 hertz. Confermando con ENTER si avrà la proiezione delle due coppie di valori da inserire nei relativi comandi SOUND.

2) Determinazione della frequenza del rumore sui canali (opzione 4): si deve inserire il valore della frequenza desiderata in output (espressa in hertz) che deve essere compreso tra 3.962 e 249.594 hertz. Confermando ENTER si avrà la proiezione della coppia di valori da inserire nel relativo comando SOUND.

3) Selezione del mix tra suono e rumore sui canali (opzione 5): si deve

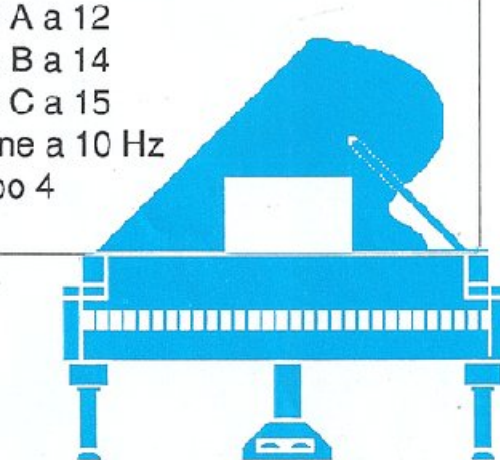
scegliere la combinazione desiderata su ogni canale in base alle sub-opzioni previste dalle tre videate intermedie (relative ai canali A, B e C). Confermando con ENTER si avrà la proiezione della coppia di valori da inserire nel relativo comando SOUND.

4) Determinazione dei volumi dei canali A, B e C (opzioni 6, 7 e 8): si deve inserire il valore della potenza del suono (e/o rumore) desiderato in output, che deve essere compreso tra 0 e 15 (parametri relativi). Confermando con ENTER si avrà la proiezione della coppia di valori da inserire nel relativo comando SOUND. E' anche possibile inserire il valore 16, che però causa una variazione del volume.

5) Selezione del tempo di modulazione del ciclo sonoro (opzione 9): si deve inserire il valore della frequenza di modulazione (espressa in hertz), che deve essere compreso tra 0,11902 (cioè quasi 12 decimi) e 15.600. Confermando con ENTER si avrà la proiezione delle due coppie di valori da inserire nei relativi comandi SOUND.

6) Scelta dell'involuppo della forma d'onda sonora (opzione 0): si deve inserire un valore (compreso tra 1 e 8) relativo alla sub-opzione preferita tra le 8 disponibili nella videata intermedia, dopodichè, confermando con ENTER, si avrà la proiezione della coppia di valori da inserire nel relativo comando SOUND. La videata introduttiva riporta anche il disegno schematico dei grafici dei vari cicli.

Poiché sono presenti limiti parametrici già stabiliti, non verranno generati valori incompatibili con i relativi registri, ma solo proiezioni completa-



mente affidabili e direttamente utilizzabili senza ulteriori modifiche o formattazioni.

Un esempio pratico

Pertanto, se si vuol generare un suono di 800 Hz sul canale A, di 2000 Hz sul canale B, e un rumore di 8320 Hz e un suono di 300 Hz sul canale C, con volumi rispettivamente a 12, 14 e 15, con un involuppo di tipo 4:



e una modulazione ciclica a 10 Hz, basterà eseguire tramite un RUN il programma riportato nella tavola 2.

Una volta eseguito, il programma continua a generare il suono risultante, indipendentemente dal resto, fintanto che non si arresta l'esecuzione con uno STOP definitivo (CTRL+STOP). Naturalmente il tutto può anche essere scritto più razionalmente come

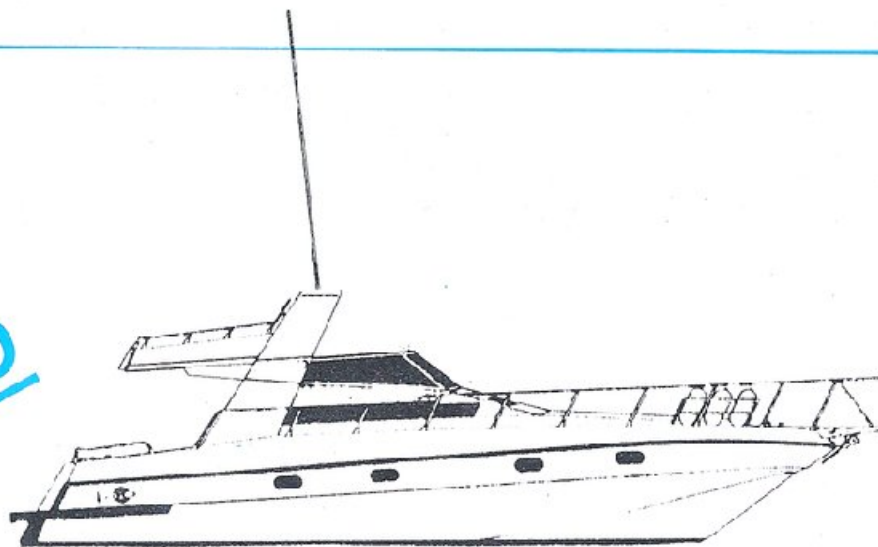
```
10 DATA 0, 156, 1, 0, 2, 62, 3, 0,
4, 160, 5, 1, 6, 15
20 DATA 7, 24, 8, 12, 9, 14, 10,
15, 11, 12, 12, 3, 13, 4
30 FOR N=1 TO 14: READ X:
READ Y
40 SOUND X, Y: NEXT
```

causando poi, in fase di esecuzione, gli stessi effetti sonori.

Daniele Malavasi
fine

NUOVO

400W SSB



NAUTICA 50W/NAUTICA 200W

Antenna ad alto rendimento, per imbarcazioni, in legno o fiberglass
Frequenza 27 MHz
Impedenza 52 Ohm
SWR: 1,2 centro banca. Antenna 1/2 lunghezza d'onda
Bobina di carico a distribuzione omogenea (Brevetto SIGMA), stilo alto cm 190 circa, realizzato in vetroresina epossidica.

MARINA 160

Frequenza 156-162 MHz
Impedenza 50 Ohm
Potenza applicabile 100 W
V.S.W.R. 1-1,1 - 1-5,1
Guadagno 3 Db (su Ground plane 1/4 d'onda)
Altezza cm. 140
Peso gr. 150
Cavo m. 0,30 RG-58U

MARINA 160 T. ALBERO

Stesse caratteristiche elettriche della Marina 160 VHF, ma corredata di supporto in acciaio inox per il montaggio a testa d'albero.

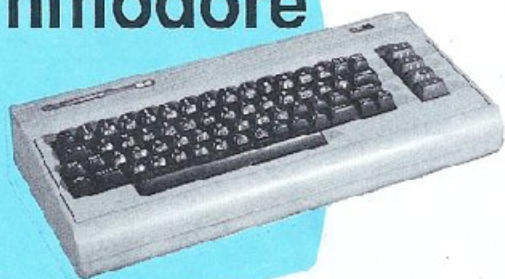
**CATALOGO A RICHIESTA
INVIANDO
L. 800 IN FRANCOBOLLI**



SIGMA ANTENNE di E. FERRARI

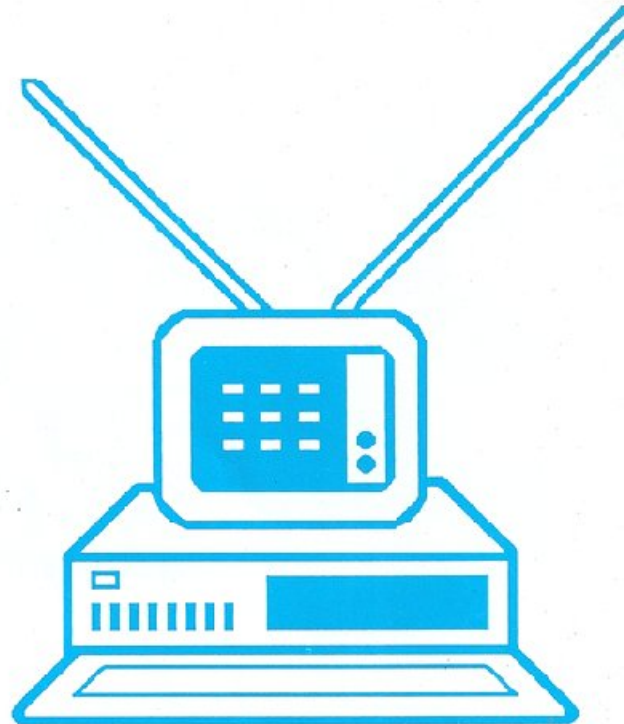
46047 S. ANTONIO MANTOVA - via Leopardi 33 - tel. (0376) 398667

**Commodore
64**



TELEVISIONE

Questa sera telecomando io



Vi piacerebbe poter scegliere tutte le sere il programma televisivo in una scaletta conforme ai vostri gusti esclusivi? Fate fatica a destreggiarvi con reti private, di stato e orari che si accavallano? Questo programma vi consentirà di ottimizzare anche le vostre ore di relax davanti al televisore.

Cosa c'è stasera in tv? Questa domanda suona piuttosto familiare nelle nostre orecchie. Forse perché ormai siamo tutti un po' videodipendenti e senza la televisione la sera non sappiamo proprio cosa fare. Spesso la televisione è solo un'abitudine, una consuetudine che riempie le poche ore che vanno dalla cena al meritato riposo, ma che rabbia sapere il giorno dopo dagli amici che ci siamo persi il film più appassionante della stagione, il reportage più interessante o lo spettacolo più esilarante.

Ancora una volta è il computer che può rimettere ordine, garantendoci una programmazione molto più efficiente. Ulteriore testimonianza della sua versatilità, ecco infatti un programma capace di pianificare perfino le nostre ore di relax davanti al televisore. E' sufficiente istruirlo una volta alla settimana e lui ci fornirà puntualmente tutte le sere gli spettacoli, gli orari, i canali, compatibilmente con le nostre esigenze e i nostri gusti.

Come funziona

TV manager è in grado di memorizzare fino a 200 spettacoli televisivi per settimana, di eseguire ricerche su tutti i campi e con condizioni multiple. Anche gli ordinamenti possono essere eseguiti in tutta libertà su uno qualsiasi dei campi. Tutti dati di una settimana, al momento dell'elaborazione devono risiedere in memoria e possono essere salvati al termine su disco o su nastro come file distinti. La settimana (dalla domenica al sabato) viene identificata dal programma con la data della domenica, tuttavia grazie a una serie di routine per l'elaborazione delle date e del calendario perpetuo, TV manager è in grado di ricondurre un qualsiasi giorno della settimana alla data della domenica corrispondente, quindi l'utente, per selezionare la settimana, può semplicemente riferirsi a un giorno qualsiasi di essa.

Tutti gli input sono rigorosamente controllati e filtrati in modo tale che record con campi scorretti non posso-

no essere registrati: il giorno della settimana per esempio deve essere un codice di tre lettere (DOM, LUN, MAR, MER, GIO, VEN, SAB), qualsiasi altra stringa di caratteri viene considerata illegale. Anche le date e le ore devono essere espresse in forme ben precise e rispettivamente in gg/mm/aa, dove gg indica il giorno del mese, mm il mese e aa l'anno, e hh.mm, dove hh indica le ore e mm i minuti. Per comodità computazionale le stringhe che definiscono le date e le ore devono essere di lunghezza fissa, quindi laddove sarebbe possibile esprimere il numero con una sola cifra è necessario aggiungere uno zero a sinistra, per esempio: 01/05/86 e 09.25. Tutta una serie di routine di controllo impediscono comunque di introdurre date e orari formalmente corrette ma prive di significato come 35/03/86, 12/13/88, 12.67, 26.45 e così via.

Da tutte le situazioni di input e da tutti i menù e sottoopzioni è possibile rinunciare e abortire l'esecuzione, tornando alla situazione di livello superiore o direttamente al menù principale con il tasto di funzione F1. Analogamente il tasto di funzione F7 ha il significato esattamente opposto e va in genere utilizzato per confermare, attivare o proseguire nelle operazioni.

Una volta avviato il programma con RUN, viene visualizzato il quadro con il menù principale (figura 1), che raccoglie tutte le opzioni di base. Ogni opzione va selezionata accendendo il reverse verde sulla riga corrispondente e attivata per mezzo della pres-

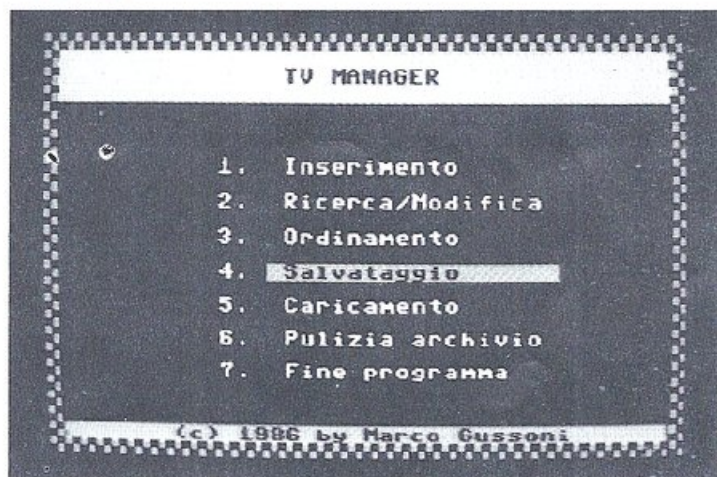


Figura 1

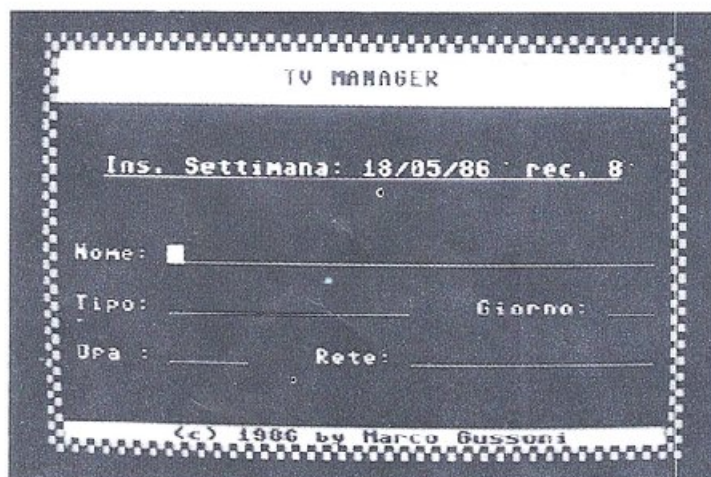


Figura 2

sione del tasto RETURN. Alcune vengono eseguite immediatamente, altre, come la 6 e la 7, chiedono conferma per evitare di distruggere delle informazioni per errore, altre ancora invece possono essere eseguite solo a certe condizioni: la ricerca, il salvataggio e l'ordinamento infatti non hanno molto senso se in memoria non sono presenti dei record. Ecco una descrizione di ogni opzione:

1. Inserimento. Serve per aggiungere nuovi elementi a quelli in memoria. Se in memoria è già presente una settimana viene chiesta conferma se si desidera effettivamente aggiungere, viceversa se l'archivio è vuoto è necessario introdurre la data della settimana a cui deve riferirsi l'archivio che si sta per approntare. Come già detto, va introdotta la data della domenica o comunque uno qualsiasi dei giorni della settimana: il programma risponderà in ogni caso con la domenica, attendendo conferma. Definita la settimana, si giunge alla maschera di input vera e propria (figura 2). A questo punto è sufficiente posizionarsi con i tasti del movimento cursore sul campo desiderato, digitare le informazioni da introdurre e quindi premere F7 per memorizzarla definitivamente e passare alla successiva o F1 per cancellarla e tornare al menù principale. Premendo F7 prima di registrarla, il programma analizza dettagliatamente tutti i campi del record, per trovare eventuali errori, nel qual caso visualizza un messaggio e costringe l'utente a correggerlo o a rinunciare all'inserimento.

2. Ricerca/Modifica. Serve per reperire un elemento all'interno dell'archivio, visualizzarlo ed eventualmente modificarlo. Gli elementi vengono ricercati per mezzo di alcuni parametri che vanno specificati nella maschera di figura 3. Possono essere poste condizioni contemporanee su tutti e 5

i campi del record, scegliendole tra le seguenti:

- = uguale a
- < minore di
- > maggiore di
- # diverso da
- * qualsiasi

Il tipo di ricerca viene selezionato anteposando all'oggetto da cercare (e come primo carattere della riga) un carattere di controllo tra quelli sopra elencati, di seguito va poi espresso il parametro vero e proprio. Un esempio chiarirà meglio le cose:

Nome: *
 Tipo: =film
 Giorno: >gio
 Ore: <21.00
 Rete: #rai 1

Con queste richieste il programma andrà a estrarre dall'archivio tutti i film, con un titolo qualsiasi, in programmazione dopo giovedì, che iniziano prima delle 21.00 e che non sono trasmessi sulla Rai 1. Per semplificare al massimo la descrizione delle richieste i simboli speciali posti a inizio riga vengono visualizzati in reverse e in colore ciano. Inoltre per consentire una più rapida editazione, premendo la sbarra spaziatrice quando il cursore si trova sul primo carattere del campo di ricerca, viene stampato il simbolo * e il cursore viene riposizionato a capo sulla riga seguente.

Una volta introdotti tutti i parametri si può procedere alla ricerca con F7 o abortire il comando con F1. A mano a mano che il programma incontra elementi che matchano con le richieste formulate vengono visualizzati sullo schermo. A questo punto è possibile scegliere se proseguire nella ricerca con il record successivo F7 o terminare anzitempo con F1.

3. Ordinamento. Consente di visualizzare, uno alla volta, tutti i record dell'archivio in memoria, ordinati in modo crescente rispetto a uno qualsiasi

dei campi. Se in memoria è presente almeno un record, viene visualizzato il sottomenù di figura 4. A questo livello è possibile selezionare il campo di ordinamento per mezzo dei tasti cursore e RETURN o tornare semplicemente al menù principale premendo F1. Selezionando uno dei campi sullo schermo appare l'indicazione del numero dei record presenti e a mano a mano quelli che già sono stati ordinati. Al termine verrà visualizzato il primo della lista. Per scorrere avanti e indietro la lista ordinata è sufficiente utilizzare i tasti CURSOR UP e RIGHT (per andare avanti) e CURSOR DOWN e LEFT (per tornare indietro). Come al solito con F1 si torna al menù principale.

4. Salvataggio. Consente di registrare su un supporto di massa l'archivio in memoria. Se sono presenti dei record, viene presentata la scelta tra disco e nastro, selezionabile premendo rispettivamente D o N, quindi si viene invitati a introdurre il nome del file destinato ad accogliere tutti i record. Nel caso si tratti di una registrazione su disco il nome del file (massimo 12 caratteri) viene completato con un suffisso ##, inoltre in caso il file fosse già presente, viene automaticamente eseguito un replace (@0:).

5. Caricamento. Consente di ricaricare in memoria i record salvati precedentemente con il comando Salvataggio. Come prima cosa il programma presenta la scelta del supporto: disco o nastro, anche in questo caso selezionabile con D e N. Successivamente viene richiesto il nome del file, se si tratta di un caricamento da disco vengono aggiunti automaticamente i due caratteri ## in coda al nome battuto dall'utente e viene controllata la correttezza dell'operazione per mezzo della lettura del canale 15. Se qualcosa non è andato bene viene visualizzato un opportuno messaggio d'avver-

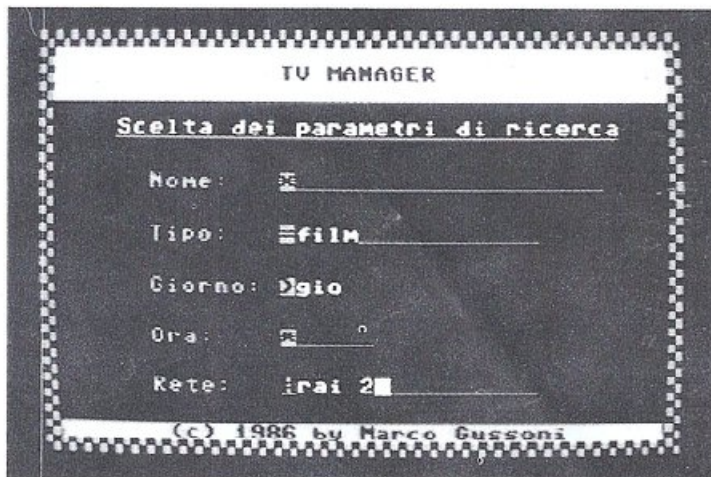


Figura 3

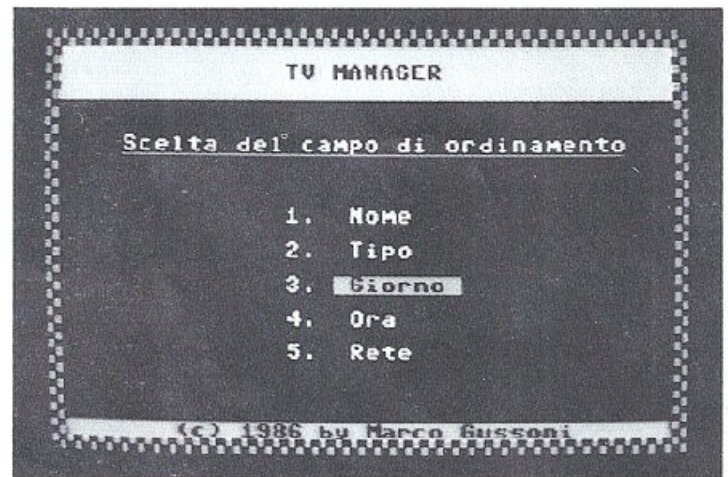


Figura 4

timento con il codice dell'errore DOS corrispondente. Se si tratta di un caricamento da nastro il sistema stesso invita a premere PLAY sul registratore e quindi il logo Commodore quando viene incontrato l'header del file. Al termine viene ricostruita la finestra di lavoro forata dai messaggi di sistema.

6. Pulizia dell'archivio. Consente di cancellare tutti i dati in memoria. Viene naturalmente chiesta conferma. Questa opzione è indispensabile se ci si accinge a digitare un nuovo archivio dopo che si è compiuta un'elaborazione precedente, tuttavia con un caricamento da disco o da nastro ciò che era presente in memoria viene automaticamente ricoperto.

7. Fine programma. Dopo la conferma pulisce lo schermo e termina il programma.

Il listato

Di questo programma si consiglia di utilizzare la versione compilata, che, oltre a una maggiore velocità nelle ricerche e negli ordinamenti, offre una più agevole editazione nelle fasi di input. Per questo motivo nella cassetta di questo mese è presente sia la versione normale (sorgente) che quella compilata (Blitz compiler). Ecco un commento alle linee del sorgente:

200-260 Fissa i colori del bordo, dello sfondo, di linea, il repeat automatico su tutti i tasti, il lower case, richiama la routine per la definizione delle variabili e salta al menù principale.

800-820 Pulisce 14 righe della finestra principale.

850-870 Pulisce 18 righe della finestra principale.

900-930 Input controllato della data.

1000-1220 Richiama la routine di definizione del bordo della finestra principale, disegna lo schermo del menù e riceve la scelta dell'utente. Per alcune

opzioni richiama la routine della conferma e per tutte salta alle linee corrispondenti.

1300-1508 Riceve l'input della data della settimana, controlla la correttezza ed eventualmente la modifica facendola coincidere con la domenica.

1510-1540 Chiede conferma della data visualizzata sullo schermo. Premendo il tasto N si annulla, un altro tasto qualsiasi accetta.

1550-1980 Disegna la maschera di input dei record, accetta i comandi dell'utente e richiama le routine di spostamento del cursore, di cancellazione del video, di inserimento e di ritorno al menù. Vengono inoltre eseguiti tutti controlli sulla legalità dei dati relativi alle date e agli orari.

2000-2120 Esegue gli ultimi controlli sulla correttezza dei dati e archivia il record in memoria. Se l'archivio è completo salta alla routine di avvertimento, altrimenti riprende con nuovo record.

2150-2180 Visualizza il messaggio il messaggio di archivio completo e riporta al menù principale.

2200 Cancella l'ultimo record puntato da NR.

2500-2530 Visualizza il messaggio E R R O R E N E I D A T I.

3000-3310 Disegna la maschera di input dei parametri di ricerca e accetta tutti comandi dell'utente. Visualizza sullo schermo i caratteri digitati, esegue lo spostamento del cursore e salta alla routine di ricerca vera e propria (F7) o al menù (F1).

3350-3505 Esegue tutti i controlli sulla correttezza delle richieste e predispone il vettore per i salti alle routine di ricerca sui campi.

3550-3650 Ciclo principale di ricerca. Vengono esaminati tutti i campi di tutti i record per vedere se matchano con le richieste. Al termine o a causa della pressione del tasto F1 si viene ricondotti al menù principale.

3900-3930 Viene visualizzato il messaggio Errore nelle richieste.

3950-3960 Visualizza sullo schermo le stringhe con le richieste.

4000-4430 Routines per la ricerca sui singoli campi.

5000-5100 Visualizza il menù di scelta del campo di ordinamento e riceve la scelta dell'utente.

5150-5230 Ordina i puntatori ai record (PT%) e visualizza il contatore.

5240-5350 Visualizza un record della lista ordinata sullo schermo e riceve i comandi di spostamento lungo la lista o di ritorno al menù.

5500-5550 Reseta la memoria e pulisce l'archivio.

5950-5965 Visualizza il messaggio Non ci sono elementi!

6000-6220 Salva l'archivio su disco o su nastro.

6500-6720 Carica l'archivio da disco o da nastro.

6800-6830 Input nome del file, per carica e salvataggio.

6900-6920 Propone la scelta tra disco e nastro.

7000-7020 Visualizza il messaggio con l'errore del DOS.

7850-7870 Input della conferma.

7900-7920 Visualizza la finestra rossa dei messaggi.

7950-7980 Disegna la finestra del quadro del menù principale.

8000-8300 Dimensiona tutti i vettori, definisce le variabili fondamentali e legge dai data le informazioni da assegnare ai vettori.

8900 Termina il programma.

9000-9130 Linee dei data con le opzioni del menù principale, i giorni dei mesi dell'anno, le posizioni del cursore per la maschera di input, i codici dei giorni della settimana, le lunghezze massime delle stringhe per le richieste delle ricerche e le opzioni per il menù di scelta del campo di ordinamento.

Marco Gussoni

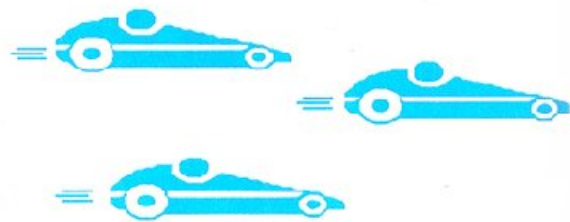
Spectrum 48K



FORMULA UNO

Per sapere chi ha vinto l'ultimo Gran Premio basta uno sguardo al giornale; ma memorizzare tutti i risultati dell'anno, tutti i nomi dei piloti e dei team, tutte le classifiche, è affare da veri amatori. Con lo Spectrum e questo programma, potete gestire un intero campionato di Formula 1.

Tutte le corse dell'anno



La gestione computerizzata del campionato mondiale piloti e costruttori della Formula 1, dal punto di vista operativo, è abbastanza complessa, perché il regolamento di questo sport è piuttosto articolato; soprattutto nella classifica piloti, la più importante delle due, sono considerati validi non già tutti i punteggi utili conseguiti nelle corse disputate, ma soltanto gli undici migliori di tutti quelli accreditati.

Accade cioè che il valore reale del punteggio totale di classifica (quello che conta ai fini del campionato, dato dalla somma degli undici migliori piazzamenti) può essere uguale oppure diverso (soltanto minore) rispetto al punteggio assoluto. Il programma Gestione Formula 1 è in grado di ovviare, in piena osservanza alle norme del regolamento vigente, a ogni tipo di difficoltà innescata da tale meccanismo, grazie alla possibilità di gestire un numero infinito di campionati, ciascuno dei quali può contenere tutti i dati relativi a ben 28 corse, 28 piloti e 20 costruttori, per un totale di oltre 3000 dati.

Oltre alle classifiche piloti e costruttori, sono disponibili altre numerose opzioni, tutte accessibili dal menù principale in modo diretto, che permettono di visualizzare gli ordini di arrivo nelle varie corse disputate e gli andamenti, sia dei piloti che dei costruttori. Sono presenti tre opzioni di aggiornamento che permettono di inserire in un qualsiasi momento nuovi costruttori e/o nuovi piloti, nonché di

aggiornare la situazione a mano a mano che nuove corse vengono disputate. Tutte le routine di input dai dati sono protette, per quanto possibile, contro ogni eventuale errore di immissione, accidentale o volontario che sia.

Ogni iter operativo è supportato da una ricca serie di strip-guida colorati che compaiono a fondo video e facilitano il lavoro.

Un efficace sistema operativo permette di scegliere, in un qualsiasi momento, se rimanere nell'ambito della stessa opzione o ritornare al menù principale. Il programma, composto da una parte principale operativa (master) e da una coppia di file di riferimento, è in grado di gestire più campionati e di essere riutilizzato per più elaborazioni semplicemente variando i file di riferimento stessi. Inoltre è possibile generarne altri semplicemente scegliendo l'opzione preliminare di accesso.

Gestione Formula 1 è stato scritto su Spectrum Plus, ma gira perfettamente su qualsiasi Spectrum anche tradizionale (Issue 1, 2, 3, eccetera) purché espanso a 48 Kbyte: infatti si spinge fino alla locazione RAM numero 63.167.

Caratteristiche del programma

Il programma è composto da quattro file consecutivi e indipendenti (tavola 1) che sono, nell'ordine di carica: sottoprogramma di lancio (basic program), screen di copertina (bytes),

generatore di grafica (bytes) e infine master (basic program). Le principali caratteristiche delle linee dei programmi basic sono le seguenti.

Sottoprogramma di lancio:

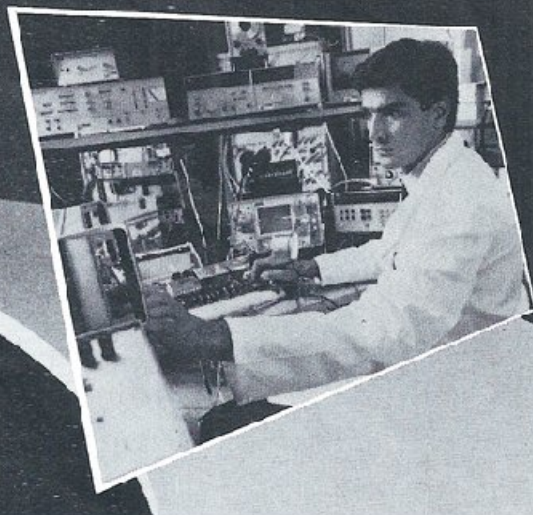
10-22 Linee di remark; settaggio video; istruzioni per il caricamento dei tre file successivi;
23-24 Data-set di copyright; stampa del titolo e del logo.

Master:

100-920 Data-set di copyright; settaggio video; preparazione della videata introduttiva; scelta della opzione di accesso e del tipo di file su cui lavorare;
960-990 Organizzazione delle finestre-video;
1001-1051 Subroutine di proiezione del menù principale; scelta della opzione operativa desiderata tra le 10 disponibili;
2101-2199 Prima opzione: aggiornamento corse (tasto 0);
2201-2299 Seconda opzione: inserimento piloti (tasto 1);
2310-2399 Terza opzione: inserimento costruttori (tasto 2);
2410-2499 Quarta opzione: classifica piloti (tasto 3);
2510-2599 Quinta opzione: classifica costruttori (tasto 4);
2610-2699 Sesta opzione: ordine di arrivo corse (tasto 5);
2710-2799 Settima opzione: andamento piloti (tasto 6);
2810-2899 Ottava opzione: andamento costruttori (tasto 7);

Fai vedere chi sei!

DIVENTA UN TECNICO IN ELETTRONICA DIGITALE E MICROCOMPUTER.



SCUOLA RADIOELETTA TI APRE LE STRADE DEL FUTURO

Se desideri assicurarti anche tu un ruolo da esperto in un modernissimo campo di attività, Scuola Radioelettra ha pronto per te il Corso-Novità ELETTRONICA DIGITALE E MICROCOMPUTER.

"IL FUTURO" PER LA TUA AFFERMAZIONE.

ELETTRONICA DIGITALE E MICROCOMPUTER. Un completo ciclo di studio al termine del quale saprai approfonditamente com'è fatto, come funziona, come si impegna e come si ripara un microcalcolatore. 46 Gruppi di Lezioni, 17 Serie di materiali. Oltre 870 componenti e accessori. Tutto è preordinato perché tu possa, a casa tua partendo dalle nozioni di base, impadronirti gradualmente e con sicurezza dei segreti dell'elettronica.



UNA GRANDE OCCASIONE PER TE.

Grazie ai materiali tecnici compresi nel Corso, fin dalle prime lezioni potrai mettere in pratica ciò che avrai imparato. **Inoltre costruirai interessanti e utilissime apparecchiature che resteranno di tua proprietà e ti serviranno sempre:**

Minilab (laboratorio di elettronica sperimentale) **Tester** (analizzatore universale) **Digilab** (laboratorio digitale da tavolo) **Eprom Programmer** (programmatore di memoria Eprom) **Elettra Computer System** (microcalcolatore basato sul microprocessore Z80).

UNA SCUOLA SU MISURA A CASA TUA

Comodità assoluta di studio, senza rinunciare alle tue attuali attività. Con Scuola Radioelettra impari come e quando vuoi tu. **con tutta l'assistenza che ti serve.**

UN METODO COLLAUDATO DAL SUCCESSO

Scuola Radioelettra mette a tua disposizione un piano di studio avanzatissimo corredato dai materiali più aggiornati che resteranno di tua proprietà.

Tutta la teoria e la pratica che serve per imparare davvero.

UNA REFERENZA INDISPENSABILE



Il tuo Attestato di studio, che a fine corso,

testimonerà il tuo livello di apprendimento.

I VANTAGGI "ELETTRACARD"

Un Club esclusivo, riservato a tutti gli Allievi Scuola Radioelettra, che ti dà diritto a tante sorprese **uniche e sempre molto vantaggiose.**

500.000 GIOVANI COME TE HANNO TROVATO LA VIA DEL SUCCESSO CON SCUOLA RADIOELETTA. ORA TOCCA A TE QUESTA GRANDE OPPORTUNITA'.

SPEDISCI SUBITO, IL TAGLIANDO RIPRODOTTO A FONDO PAGINA, RICEVERAI GRATIS E SENZA IMPEGNO TUTTE LE INFORMAZIONI CHE DESIDERI.

CON SCUOLA RADIOELETTA PUOI SCEGLIERE FRA 30 OPPORTUNITA' PROFESSIONALI.

Corsi di Elettronica

- Tecnica elettronica sperimentale
- ▶ Elettronica fondamentale e telecomunicazioni
- ▶ Elettronica digitale e microcomputer
- ▶ Parla Basic
- Elettronica industriale
- Elettronica televisione
- Televisione bianco e nero
- Televisione a colori
- Amplificazione stereo
- Alta fedeltà
- Strumenti di misura

Corsi Tecnico Professionali

- ▶ Elettrotecnica
- Disegnatore meccanico progettista
- Assistenza e disegnatore edile
- Motorista autoriparatore
- Tecnico d'officina
- Elettraulico
- Programmazione su elaboratori elettronici
- ▶ Impianti a energia solare
- ▶ Sistemi d'allarme antifurto
- ▶ Impianti idraulici-sanitari

Corsi Commerciali

- Esperto commerciale
- Impiegata d'azienda
- Dattilografa
- Lingue straniere

Corsi Professionali e Artistici

- ▶ Fotografia bianco e nero
- ▶ Fotografia stampa del colore
- ▶ Disegno e pittura
- ▶ Esperta in cosmetici
- ▶ Cucito a macchina

▶ **Questo simbolo indica i CORSI NOVITA'.**

Fresa d'atto del Ministero della Pubblica Istruzione n. 1391

Scuola Radioelettra è associata alla A.I.S.CO (Associazione Italiana Scuole per Corrispondenza per la tutela dell'allievo).



Scuola Radioelettra

VIA STELLONE 5, 10126 TORINO, TEL. (011) 674432

Per te

XE 87

Si, Vi prego di farmi avere, gratis e senza impegno, il materiale informativo relativo al

Corso di: _____

COGNOME _____

NOME _____

VIA _____ N° _____

LOCALITA' _____

CAP _____ PROV. _____ TEL. _____

ETA _____ PROFESSIONE _____

MOTIVO DELLA RICHIESTA: PER LAVORO PER HOBBY

Per un tuo amico

XE 88

Si, Vi prego di farmi avere, gratis e senza impegno, il materiale informativo relativo al

Corso di: _____

COGNOME _____

NOME _____

VIA _____ N° _____

LOCALITA' _____

CAP _____ PROV. _____ TEL. _____

ETA _____ PROFESSIONE _____

MOTIVO DELLA RICHIESTA: PER LAVORO PER HOBBY

CON NOI PUOI

TAVOLA 1. Caratteristiche dei file

N.	Nome	Tipologia Funzione	Istruzioni per la memorizzazione	Tempo carica
1	239C	Basic program sottoprogramma lancio	SAVE"239C"LINE 1	12 sec
2	COVER	bytes screen di copertina	SAVE"COVER"SCREEN\$	42 sec
3	CHR	bytes generatore grafica	SAVE"CHR"CODE\$2144,1024	14 sec
4	239P	Basic program Master	SAVE"239P" LINE112	176 sec

2910-2999 Nona opzione: memorizzazione dati (tasto 8);
 3010-3020 Decima opzione: fine lavoro (tasto 9);
 8010-8099 Memorizzazione delle variabili principali permanenti;
 8010-8199 Subroutine di CLS parziale indirizzato dal campo-video;
 8210-8299 Subroutine di proiezione indirizzata dei messaggi scorrevoli di informazione;
 8310-8399 Subroutine di proiezione indirizzata e centrata dei testi;
 8410-8499 Subroutine di attesa e di scelta del tipo di opzione di ritorno (tasti <SPACE> e <ENTER>);
 8510-8599 CLS parziale del fondo video;
 8610-8699 Inserimento dei nominativi richiesti in input;
 8720-8799 Stampa indirizzata dei testi;
 8810-8899 Sequenza musicale di segnalazione;
 8910-8999 Calcolo dell'incremento positivo dei punteggi;
 9010-9099 Conversione codificata delle variabili di controllo dei nominativi non piazzati (NC) o non partiti (NP);
 9105-9199 Algoritmo di gestione degli elenchi di classifica;
 9210 e 9299 Subroutine di blocco di accesso alle opzioni di visualizzazione;
 9994-9999 Data-set di copyright; stampa del logo.

Funzionamento

Il programma parte automaticamente dopo un periodo di carica di 246 secondi: a questo tempo va eventualmente aggiunto il periodo supplementare di 74 secondi richiesto da file di riferimento già caratterizzato che viene cari-

cato a programma già iniziato. Terminato il caricamento, compare il messaggio scorrevole a centro video "fermare il nastro, per iniziare premere <ENTER>". Poi è necessario specificare se si desidera operare su un file di riferimento di tipo vergine (tasto 1) oppure già caratterizzato (tasto 2). Nel primo caso si accederà direttamente al menù principale delle 10 opzioni operative e si creerà una nuova gestione, altrimenti si caricherà il file di riferimento.

Ovvio che la prima scelta si effettua quando si inizia la gestione di un nuovo campionato di Formula 1, mentre la seconda opzione va richiamata in ogni successiva attività di aggiornamento o modifica del file così iniziato e caratterizzato: la carica di un file già caratterizzato richiederà altri 74 secon-

TAVOLA 2. Elenco dei costruttori o team

Nome del team	Abbreviazione
Arrows	(ARR)
Benetton	(BEN)
Brabham	(BRA)
Ferrari	(FER)
Ligier	(LIG)
Lola Force	(LOL)
Lotus	(LOT)
Mc. Laren	(MC.)
Minardi	(MIN)
Osella	(OSE)
Ram	(RAM)
Tyrrell	(TYR)
Williams	(WIL)
Zakspeed	(ZAK)

di di tempo, e va eseguita alla comparsa dell'apposito strip di avviso di colore blu, che invita a far partire il registratore in <PLAY>.

Per selezionare una delle dieci opzioni operative disponibili, che sono indicate in colore verde, si usa l'apposito tasto numerico di selezione (da 0 a 9) sulla sinistra della videata. Non appena la selezione viene decisa sparisce il menù e in alto al centro compare il titolo (strip di colore blu) dell'opzione richiamata, che rimarrà fino alla fine, quando un'apposita routine di attesa permetterà di decidere se rimanere nell'ambito della stessa opzione (tasto <SPACE>) o se ritornare al menù principale (tasto <ENTER>).

Opzioni operative

Sono disponibili 10 opzioni: le prime tre di modifica e aggiornamento di corse, piloti e costruttori (tasti di accesso 0, 1 e 2). Altre cinque permettono l'analisi e la visualizzazione dei dati inseriti, proiettando classifiche ufficiali di piloti e costruttori, ordini di arrivo di particolari corse, oppure tutti gli andamenti statistici sia dei piloti sia dei costruttori (tasti di accesso 3, 4, 5, 6 e 7). Le altre due opzioni sono relative alla memorizzazione dei dati inseriti di volta in volta (tasto di accesso 8) e alla specifica del fine-lavoro (tasto di accesso 9).

• Prima opzione: aggiornamento corse.

Si richiama con il tasto 0 e può essere utilizzata solo a condizione che siano stati in precedenza memorizzati nominativi di piloti; in caso contrario compare a fondo video il messaggio "nessun pilota registrato, nominativo non inseribile". Procedendo invece regolarmente si otterrà la proiezione aggiornata del numero di corse registrate e di quelle ancora inseribili, che possono essere al massimo 28 (più che sufficienti per un normale campionato di Formula 1). Viene anche visualizzato l'elenco dei nominativi di tutte le corse eventualmente già registrate in precedenza. Occorre quindi premere <ENTER> e inserire il nominativo della corsa da memorizzare, che sarà quello ufficiale attribuito di consuetudine. Ad esempio per il Gran Premio d'Italia si inserirà il nominativo ITALIA. Vengono accettati nominativi con un numero massimo di 13 caratteri alfanumerici (oltre a spazio e punto). La parola verrà formata sullo strip giallo che compare nella parte bassa sulla sinistra del video. Eventuali errori di input potranno essere corretti con <DELETE> (nello Spectrum tradizionale è

TAVOLA 3.
Elenco dei piloti

Nome del pilota	Team di appartenenza
Alboreto	(Ferrari)
Alliot	(Ram)
Arnoux	(Ligier)
Berger	(Benetton)
Boutsen	(Arrows)
Brundle	(Tyrrell)
Capelli	(Osella)
Dumfries	(Lotus)
Fabi	(Benetton)
Johansson	(Ferrari)
Jones	(Lola Force)
Laffite	(Ligier)
Mansell	(Williams)
Martini	(Minardi)
Nannini	(Minardi)
Palmer	(Zakspeed)
Patrese	(Brabham)
Piquet	(Williams)
Prost	(Mc. Laren)
Rosberg	(Mc. Laren)
Senna	(Lotus)
Surer	(Arrows)
Streiff	(Tyrrell)
Tambay	(Lola Force)

richiamabile premendo contemporaneamente <CAPS SHIFT> e <0>. Confermando inserimenti nulli (privi cioè di caratteri) è possibile evitare l'inserimento e portarsi direttamente alla routine finale di attesa senza che nulla venga modificato o aggiornato nel file di riferimento: questa possibilità è utile in tutti i casi in cui si accede per errore all'opzione di aggiornamento. La conferma va data premendo <ENTER>, dopodiché il nominativo della corsa così inserito viene aggiunto (in colore magenta) a quelli già presenti nell'elenco della videata, e proiettato in strip verde lampeggiante in alto al centro dello schermo. A questo punto l'elenco-corse scompare e uno strip rosso a fondo video invita a scegliere il meccanismo di attribuzione del punteggio da applicare nella corsa inserita, che può essere normale (e cioè 9 punti al vincitore, 6 al secondo classificato, 4 al terzo, 3 al quarto, 2 al quinto e 1 punto al sesto) oppure dimezzato se la corsa, come dice il regolamento, è stata sospesa per cause di forza maggiore, come gravi incidenti o maltempo, dopo un numero di giri superiore alla metà di quelli previsti

alla partenza. La scelta va fatta premendo <ENTER> (oppure <SPACE> in questa seconda eventualità). Segue la proiezione dell'elenco dei piloti presenti: dando <ENTER> si accede alla routine di attribuzione dei relativi piazzamenti, da specificare per ogni nominativo (via via indicato) componendo un numero compreso tra 1 (per il vincitore primo piazzato) e un massimo di 28, oppure scrivendo NC se il pilota non si è classificato o NP se non è nemmeno partito in quanto non qualificato nelle prove preliminari di selezione. Data la conferma con <ENTER>, il piazzamento viene visualizzato a fondo blu sulla destra di ogni nominativo.

• **Seconda opzione: inserimento piloti.**

Si richiama con il tasto 1 e può essere utilizzata solo se sono stati in precedenza inseriti nominativi di costruttori; in caso contrario compare a fondo video il messaggio "nessun costruttore registrato, nominativo non inseribile". Procedendo invece regolarmente, si otterrà la proiezione aggiornata del numero dei piloti inseriti e di quelli ancora inseribili, che possono essere al massimo 28 (anche in questo caso più che sufficienti per un normale campionato di Formula 1). Viene anche visualizzato l'elenco con i nominativi di tutti i piloti (e delle abbreviazioni delle relative marche dei costruttori di appartenenza) eventualmente già memorizzati in precedenza. Occorre poi premere <ENTER> e inserire il nominativo del pilota da registrare: ad esempio per il pilota Michele Alboreto si inserirà il nominativo ALBORETO. Vengono accettati nominativi con un numero massimo di 10 caratteri alfanumerici (oltre a spazio e punto), che possono essere composti da tastiera. Confermando inserimenti nulli (privi cioè di caratteri) è possibile evitare l'inserimento e portarsi direttamente alla routine finale di attesa. La conferma va data con <ENTER>, dopodiché il nominativo del pilota così inserito viene aggiunto (in colore magenta) a quelli già presenti nell'elenco della videata, e proiettato nella finestra di servizio a fondo video. Occorre quindi, come avverte un apposito strip scorrevole rosso, associare al nominativo del pilota quello del relativo costruttore di appartenenza: a tal fine l'elenco dei piloti viene sostituito con quello dei costruttori; ogni nominativo è preceduto da una lettera di colore magenta che indica il tasto da premere per effettuare l'associazione pilota-costruttore. La scelta va effettuata premendo <ENTER> e quindi il tasto di

abbinamento, dopodiché compare (in rosso a fondo video) il costruttore associato al nominativo del pilota inserito, e viene visualizzato l'elenco aggiornato dei piloti (con le rispettive abbreviazioni dei nominativi dei costruttori di appartenenza).

• **Terza opzione: inserimento costruttori.**

Si richiama con il tasto 2 ed è quella da utilizzare per prima, in quanto consente poi di accedere a tutte le altre. All'inizio si ottiene la proiezione aggiornata del numero dei costruttori già inseriti e di quelli ancora memorizzabili, che possono essere al massimo 20 (un normale campionato di Formula 1 non ha mai in media più di una quindicina di marche o team). Viene anche visualizzato l'elenco di tutte le marche già memorizzate. Si preme quindi <ENTER> e si compone il nominativo del nuovo costruttore da inserire in elenco, che corrisponderà al nome ufficialmente noto: ad esempio per il team Lotus si inserirà il nominativo LOTUS. Vengono accettate immissioni con un numero massimo di 10 caratteri alfanumerici (oltre a spazio e punto) che possono essere composte in modo sequenziale da tastiera.

• **Quarta opzione: classifica piloti.**

Va richiamata con il tasto 3 e si può utilizzare solo a condizione che

TAVOLA 4.
Calendario ufficiale

Gran Premi previsti

- Brasile (23 marzo)
- Spagna (13 aprile)
- San Marino (27 aprile)
- Monaco (11 maggio)
- Belgio (25 maggio)
- Canada (15 giugno)
- U.S.A. (22 giugno)
- Francia (6 luglio)
- Gran Bretagna (13 luglio)
- Germania Federale (27 luglio)
- Ungheria (10 agosto)
- Austria (17 agosto)
- Italia (7 settembre)
- Portogallo (21 settembre)
- Messico (12 ottobre)
- Australia (26 ottobre)
- Sudafrica (8 novembre)

siano stati precedentemente inseriti i dati relativi a Gran Premi già corsi. In caso contrario compare a fondo video il messaggio "nessuna corsa registrata, proiezione non visualizzabile". Procedendo correttamente si ottiene invece la proiezione della classifica ufficiale aggiornata di tutti i piloti inseriti, in base alla sommatoria dei punteggi ottenuti nelle varie corse disputate (in cui si siano realizzati piazzamenti in zona punti, ovvero dal 1° al 6° posto). Il nome del pilota è integrato dalla sigla abbreviata in color magenta del team di appartenenza, dal punteggio assoluto, indicato suilla destra in blu e tra parentesi, e dal punteggio reale, in rosso, che è quello importante ai fini della vincita del campionato (è la somma degli undici migliori punteggi fino al momento ottenuti, al netto di tutti quelli scartati), che può essere uguale o minore di quello assoluto in base alle situazioni che di volta in volta si creano. E' già prevista la routine di calcolo e indicazione di due o più eventuali piloti a pari merito, con i conseguenti spostamenti dei nominativi a posizione inferiore. L'opzione è composta di due pagine video, la prima indica le prime 14 posizioni, la seconda le rimanenti inferiori fino alla 28. Qualora siano presenti in elenco più di 14 piloti si può accedere alla seconda videata con il tasto <ENTER>.

• Quinta opzione: classifica costruttori.

Va richiamata con il tasto 4. Vale tutto quanto detto a proposito della classifica piloti (terza opzione), solo che l'elenco visualizzato è quello dei costruttori, che viene redatto in base al punteggio assoluto accumulato nel corso delle varie gare da tutti i piloti piazzati appartenenti allo stesso team a cui sono stati abbinati in fase di codifica: per la classifica costruttori il regolamento di assegnazione dei punti non prevede infatti la scelta di un numero limitato degli undici migliori punteggi. Anche in questa opzione è presente la seconda pagina video richiamabile con il tasto <ENTER>.

• Sesta opzione: ordine di arrivo corse.

Permette di conoscere l'esatto ordine di arrivo dei primi 6 piloti classificati (piazzati in zona punti) nell'ambito del Gran Premio scelto tra quelli registrati. Si richiama con il tasto 5 e può essere utilizzata solo a condizione che siano stati precedentemente inseriti i dati relativi a corse già disputate; in caso contrario compare a fondo

video il messaggio "nessuna corsa registrata, proiezione non visualizzabile". Procedendo correttamente si ottiene invece la proiezione dell'elenco delle corse già registrate, ciascuna preceduta da un numero (in colore magenta) che indica il codice di accesso da comporre nello strip giallo della finestra di servizio (dopo aver dato <ENTER>) per scegliere la corsa di cui visualizzare i piloti piazzati. Il codice va composto usando i tasti numerici da 0 a 9. Confermando l'input con <ENTER> l'elenco delle corse scompare e viene visualizzata la videata comprendente lo strip verde lampeggiante con il nome del Gran Premio scelto e l'elenco dei piloti piazzati (compresi il team di appartenenza, il piazzamento realizzato ed il relativo punteggio).

• Settima opzione: andamento piloti.

Permette di conoscere la proiezione storico-statistica dei punteggi ottenuti da un pilota prescelto nell'ambito delle varie corse disputate. Si ottiene la visualizzazione dell'elenco dei piloti già registrati, ciascuno preceduto da un numero (in colore magenta) che indica il codice di accesso da comporre nello strip giallo della finestra di servizio (dopo aver dato <ENTER>) per selezionare il pilota di cui analizzare l'andamento nelle varie corse disputate (in base ai punti via via guadagnati). Il codice va composto usando i tasti numerici (da 0 a 9). Confermando con <ENTER> l'elenco dei piloti scompare e uno strip lampeggiante a fondo video indica il nominativo prescelto. Fa seguito la proiezione di tutte le corse registrate con l'indicazione, per ogni nominativo, dei punti conseguiti (in colore rosso).

• Ottava opzione: andamento costruttori.

Permette di conoscere la proiezione storico-statistica dei punteggi ottenuti da un costruttore prescelto nell'ambito delle varie corse disputate dai piloti ad esso abbinati. Si richiama con il tasto 7.

• Nona opzione: memorizzazione dati

Si richiama con il tasto 8 e permette di registrare il file di riferimento (vergine o già caratterizzato in precedenti occasioni) in modo da memorizzare l'insieme delle operazioni di aggiornamento su di esso svolte, che potranno essere riutilizzate per ogni successiva eventuale modifica di aggiornamento. Uno strip blu scorrevole a fondo video invita a predisporre il regi-

stratore sulla funzione di memorizzazione (REC) e a premere il tasto <ENTER>, dopodiché il salvataggio del blocco (composto in realtà da due file) avrà automaticamente inizio, come indica il messaggio lampeggiante "registrazione in corso". L'operazione di trasferimento dura 74 secondi, e a essa deve far seguito quella di verifica, sempre lunga 74 secondi: ovviamente è necessario riavvolgere il nastro e predisporre il registratore in esecuzione (PLAY). Se la verifica viene correttamente eseguita compare il messaggio "verifica eseguita, tutto OK", cui fa seguito l'usuale routine di attesa a fine opzione. Il blocco così caratterizzato va caricato all'occorrenza all'inizio del programma (tasto 2 del menù introduttivo di accesso) nell'eventualità di successive operazioni di aggiornamento e/o di modifica dai dati.

• Decima opzione: fine lavoro.

Si seleziona premendo il tasto 9 e non ha funzione operativa, ma solamente di segnalazione dell'avvenuta decisione di fine lavoro. Va ovviamente richiamata solo all'occorrenza in quanto non è dotata della consueta routine di attesa e di ritorno presente nelle altre opzioni, ma è in pratica un vicolo cieco.

Campionato 1986

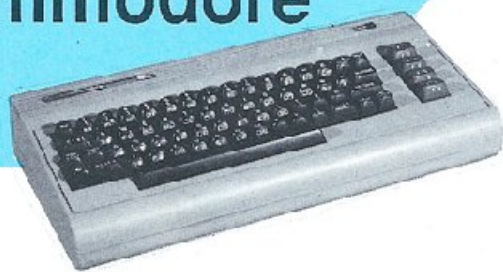
Tutti i giornali sportivi riportano, almeno nelle pagine del lunedì successivo alle domeniche in cui si disputa un Gran Premio, esaurienti elenchi dei nominativi di corse, piloti e relativi costruttori-team di appartenenza. Per chi desidera comunque seguire il campionato in corso (1986) è opportuno ricordare i nomi dei 14 team ufficialmente iscritti al campionato, da inserire servendosi della terza opzione del programma (tasto 2), che sono visibili nella tavola 2 (tra parentesi è indicato il nominativo abbreviato che verrà di solito visualizzato).

24 risultano essere invece i piloti attualmente in lizza. Possono essere inseriti tramite la seconda opzione del programma (tasto 1), e sono visibili nella tavola 3 (tra parentesi è indicato il team di appartenenza).

Tramite la prima opzione (tasto 0) vanno quindi inseriti (ma solo a corsa già disputata e con ordine di arrivo alla mano) i 17 Gran Premi previsti, che sono visibili nella tavola 4; tra parentesi è indicata la data del 1986 in cui, salvo contrattempi o cause di forza maggiore, si disputano secondo il calendario ufficiale.

Daniele Malavasi

Commodore 64



PROGRAMMAZIONE

Realizzare un arcade in tempo reale?
Senza però passare mesi di duro lavoro alla tastiera e senza conoscere il linguaggio macchina? E' sempre mancato fino a oggi, ma ora è arrivato: un software specifico per la creazione professionale di giochi.

Fate il vostro gioco

Chi possiede un C64 certamente ha avuto modo di rendersi conto delle sue straordinarie capacità grafiche e della potenza e versatilità degli effetti sonori. Anche chi solitamente lo utilizza come strumento di lavoro ha potuto ammirare gli effetti spettacolari che il C64 riesce a produrre nei videogame dell'ultima generazione. Persino alcune delle più famose avventure cinematografiche sono state ridotte e riarrangiate per essere implementate su questo computer: Ghostbuster, Rambo, Indiana Jones, eccetera.

Se da un lato, però, fa piacere sapere che il proprio computer è capace di tutto questo, dall'altro resta l'amara constatazione che simili capolavori di programmazione siano riservati solo a pochi geni del linguaggio macchina. Realizzare un videogame che sia competitivo con quelli d'oltre oceano (gli arcade in tempo reale per intenderci) crea una serie di problemi tecnici che solitamente richiedono mesi di duro lavoro alla tastiera e naturalmente una padronanza assoluta del linguaggio macchina del 6502 e di tutte le routine del kernal del 64. Anche dando per scontate le conoscenze necessarie resterebbero sempre quei "mesi alla tastiera" che per un solo programma sembrano un po' troppi.

Quello che è sempre mancato fino a oggi è stato un software specifico per la creazione di giochi. Le espansioni grafiche, i sintetizzatori, i compositori, eccetera, in realtà non risolvono il problema del progettista di giochi, perché, pur consentendo alcuni passi avanti nella definizione delle varie componenti: immagini (sprite e caratteri

di sfondo), effetti sonori (esplosioni, spari, scontri) e temi musicali, non permettono di assemblare insieme agevolmente tutti i pezzi e soprattutto non dispongono delle macro indispensabili per sopperire alla lentezza del linguaggio interpretato.

Oggi finalmente questo strumento esiste; si chiama Game Maker, è prodotto e distribuito dalla Activision e realizzato da Garry Kitchen. La versione italiana realizzata e distribuita da Stregatto, Corso di Porta Romana 55, Milano, comprende anche un manuale in italiano.

Come si usa

Si tratta di un pacchetto integrato di ben 5 programmi, completo di una vasta libreria di immagini, di suoni e di musiche. Ecco il menù principale che raggruppa tutti i tools del pacchetto:

1. Editor
2. SceneMaker
3. SpriteMaker
4. SoundMaker
5. MusicMaker

L'editor è l'ambiente di assemblaggio delle varie componenti e di stesu-



Figura 1. Una scenografia costruita con SceneMaker.

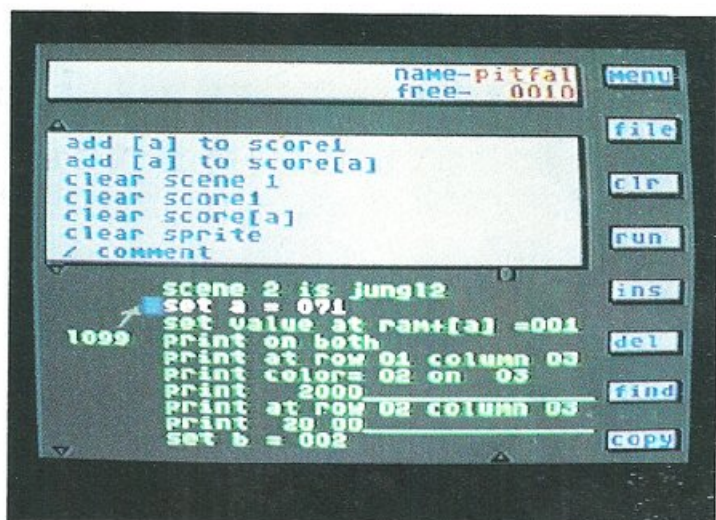


Figura 2. Piano di lavoro dell'Editor.

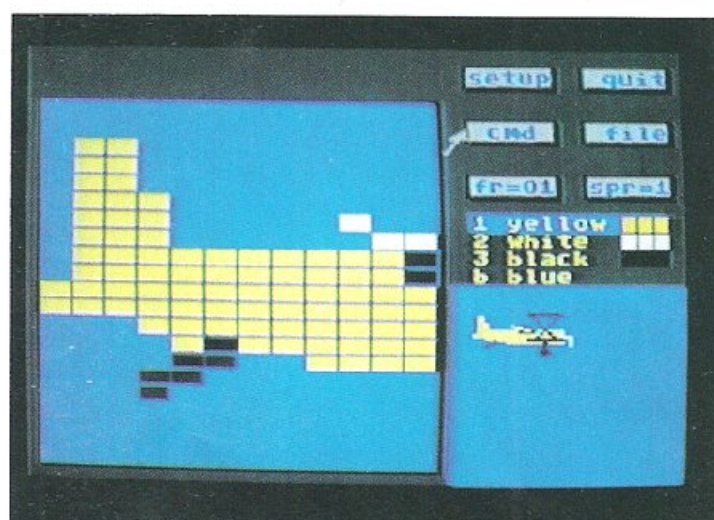


Figura 3. Biplano a due sprite costruito con SpriteMaker.

ra del codice del gioco stesso. In questa fase vengono organizzate e gestite tutte le entità costruite con i vari tools e viene definita la meccanica del gioco per mezzo di un vero e proprio linguaggio ad altissimo livello e specifico per la costruzione di giochi. Le restanti opzioni richiamano programmi che servono per la messa a punto degli oggetti: sfondi e scenari, sprite e

figure in movimento e animate, effetti sonori e temi musicali. Ognuno di questi sottoprogrammi salva il proprio lavoro su un file su disco che viene poi elaborato e linkato a tutto il resto per mezzo delle macroistruzioni dell'editor. Una volta che il gioco è stato testato in ogni sua parte, l'opzione Make-a-Disk dell'editor consente di creare una versione a se stante che po-

trà girare indipendentemente da GameMaker. Il pacchetto è predisposto per essere utilizzato con un joystick e quindi le scelte e i comandi vanno eseguiti posizionando il puntatore per effettuare la selezione e premendo il pulsante di fuoco per attivarla. Tuttavia un uso combinato dei tasti funzione e della barra spaziatrice può dare gli stessi risultati. Veniamo ora alla descrizione

TAVOLA 1

```

add 0000 to score 1
add 0000 to score [a]
add [a] to score 1
add [a] to score [a]
clear scene 1
clear score 1
clear score [a]
clear sprite
/ comment
data table at 1001
data values 000 000
display other scene
display scene 1
endif
if...then
if button 1 is on then
if joystick 1 is right
if score 1 > 000000 then
if score [a] > 000000 then
if score 1 > score 2 then
if sprite hit sprite then
jump to label 1001
jump to label [a]
jump to subroutine at 1001
jump to subroutine at [[a]
otherwise
pause for 00.0 seconds
plot a dot at x = 000 y = 000
    
```

somma un valore tra 10 e 1000 allo score 1 o 2
 somma un valore tra 10 e 1000 allo score specificato da [a]
 somma un valore [a] allo score 1 o 2
 somma un valore [a] allo score specificato da [a]
 cancella le informazioni di una scena e porta i colori in nero
 setta a 0 lo score indicato e lo visualizza sullo schermo
 setta a 0 lo score specificato da [a] e lo visualizza sullo schermo
 rende invisibile uno sprite posizionandolo fuori dallo schermo
 permette di introdurre fino a 25 caratteri di commento
 indica al computer dove si trova la tabella di data corrente
 mette i valori nella tavola dei data
 mostra sul video l'altra scena (tra le due possibili)
 mostra sullo schermo la scena specificata
 individua la fine della funzione logica if
 individua l'inizio della struttura di controllo if
 controlla se il pulsante di fuoco del joystick è stato premuto
 controlla la posizione del joystick (right, left, up, down e off)
 controlla se lo score specificato è maggiore di un certo valore
 controlla se lo score [a] è maggiore di un certo valore
 controlla se uno score è maggiore dell'altro
 controlla se sono avvenute collisioni di sprite
 fa saltare il programma ad una determinata label
 fa saltare il programma alla label specificata dalla variabile [a]
 fa saltare alla subroutine della label indicata
 fa saltare alla subroutine della label nella variabile [a]
 indica al programma l'alternativa da eseguire se l'if then risulta falso
 causa una pausa nell'esecuzione del programma
 accende un pixel in una determinata posizione della scena



Figura 4. Sintetizzatore di SoundMaker.

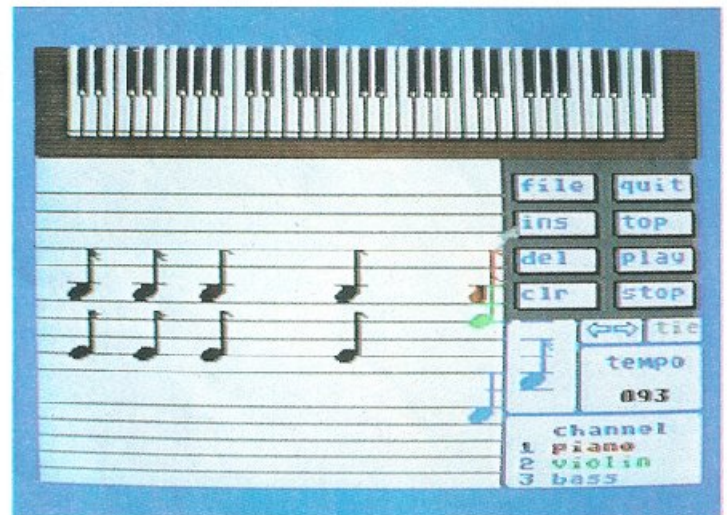


Figura 5. Spartito musicale editato con MusicMaker.

ne più approfondita di tutti gli ambienti di sviluppo.

• **L'editor.** In questa fase lo schermo si presenta diviso in 4 zone distinte (figura 2):

1. Sulla destra appaiono le opzioni di editing: MENU per selezionare le altre opzioni del pacchetto, FILE per effettuare operazioni con i file sul di-

sco, RUN per far girare il programma in memoria e INS, DEL, FIND e COPY per manipolare le linee del programma. Qui, come altrove, è sufficiente posizionare il puntatore sul rettangolo dell'opzione desiderata e premere il pulsante del fuoco.

2. La zona superiore è riservata ai messaggi, che possono essere di errore, di avvertimento o informativi.

3. La zona centrale è la finestra dei comandi e consente di scorrere bidirezionalmente tutto l'elenco dei comandi disponibili nel linguaggio di implementazione del gioco, di selezionare quello desiderato e di portarlo nel listato.

4. La zona inferiore costituisce l'area di programma e visualizza una porzione del codice del gioco in costruzio-

TAVOLA 2

<p>plot a dot at x = [a] y = [a] plot color 0 to scene 1 plot color [a] to scene 1 print print at row 00 column 00 print at row [a] column [a] print character of [a] print color = 00 on 00 print color = [a] on [a] print on scene 1 print value of [a] return from subroutine scene 1 background = black scene background 1 = [a] scene 1 border = black scene 1 border = [a] scene 1 color 1 = black scene 1 color 1 = [a] scene 1 is [] scene 2 color 1 = black scene 2 color 1 = [a] score 1 at row 00 column 00 score 2 at row 00 column 00 score 1 color = 00 on 00 score 2 color = 00 on 00 score 1 display on scene 1 screen update on/off set a = 000</p>	<p>accende un pixel nella posizione specificata dalle variabili [a] designa il colore da utilizzare in un comando plot su una certa scena designa il colore per il comando plot contenuto nella variabile [a] stampa sullo schermo 20 caratteri nella posizione selezionata da print at indica al computer la posizione della print successiva designa la posizione della print per mezzo delle variabili [a] stampa il carattere specificato dalla variabili [a] designa il colore di linea e di sfondo per i caratteri che verranno stampati designa gli stessi colori ma per mezzo delle variabili [a] designa su quale scena devono essere fatte le print stampa il valore numerico a tre cifre di una variabile chiude una subroutine seleziona il colore dello sfondo per la scena seleziona il colore dello sfondo in base al contenuto della variabile [a] seleziona il colore del bordo di una scena seleziona il colore del bordo di una scena in base al contenuto della variabile [a] seleziona i colori della scena 1 trascurando quelli utilizzati in SceneMaker seleziona i colori della scena 1 in base al contenuto della variabile [a] carica una scena in memoria seleziona i colori per la scena 2 trascurando quelli utilizzati in SceneMaker seleziona i colori della scena 2 in base al contenuto della variabile [a] designa la posizione di visualizzazione dello score 1 designa la posizione di visualizzazione dello score 2 designa il colore di linea e di sfondo dello score 1 designa il colore di linea e di sfondo dello score 2 seleziona in quale scena lo score designato deve apparire fa in modo che tutti i cambiamenti sul video vengano mostrati istantaneamente setta il valore di una variabile con un numero tra 0 e 255</p>
--	--

TAVOLA 3

set a = [a]
 set a = rnd number from 0 to 000
 set a = sprite x position
 set a = sprite y position
 set a = value at data + [a]
 set a = value at ram + [a]
 set value at ram + [a] = 000
 set value at ram + [a] = [a]
 skip next if a = 000
 song is []
 song volume = 00
 sound channel 1 = []
 sound channel 1 off
 sprite 1 is []
 sprite animates always
 sprite animation spd = 000
 sprite animation spd = [a]
 sprite color 1 = black
 sprite color 1 = [a]
 sprite dir = 000
 sprite dir = [a]
 sprite movement speed = 000
 sprite movement speed = [a]
 sprite shared colr2 = black
 sprite under colrs 2/3
 sprite x position = 000
 sprite x position = [a]
 sprite y position = 000
 sprite y position = [a]
 stop program
 trace of [a] on

setta una variabile uguale al valore di un'altra
 assegna ad una variabile un numero casuale tra 0 e il numero specificato
 assegna alla variabile la posizione x dello sprite selezionato
 assegna alla variabile la posizione y dello sprite selezionato
 assegna alla variabile il valore della tavola puntato da [a]
 assegna alla variabile il valore della tavola di ram puntata
 setta l'elemento della tavola ram puntato uguale al numero indicato
 setta l'elemento della tavola ram puntato uguale al valore della variabile
 salta l'istruzione successiva se l'espressione è vera
 carica la canzone in memoria
 seleziona il volume della musica
 seleziona l'effetto sonoro che deve essere eseguito da un certo canale
 spegne il suono su un determinato canale
 definisce lo sprite selezionato
 indica che l'animazione deve proseguire ciclicamente e continuamente
 seleziona la velocità di animazione di un certo sprite
 seleziona la velocità di animazione uguale al contenuto della variabile [a]
 designa il nuovo colore 1 per lo sprite selezionato
 designa il nuovo colore 1 uguale al contenuto della variabile [a]
 setta la direzione di movimento dello sprite
 setta la direzione uguale al contenuto della variabile [a]
 definisce la velocità di spostamento dello sprite
 definisce la velocità di spostamento uguale al contenuto della variabile [a]
 Setta il colore di sprite condiviso da tutti e 8
 seleziona la priorità dello sprite rispetto alla scena
 setta la posizione orizzontale dello sprite
 setta la posizione orizzontale uguale al valore della variabile [a]
 setta la posizione verticale dello sprite
 setta la posizione verticale uguale al valore della variabile [a]
 interrompe l'esecuzione del programma
 visualizza sul fondo il testo del comando mentre viene eseguito

ne. Come la finestra dei comandi può essere scollata in entrambe le direzioni. Con un editor e un linguaggio così sofisticati, i numeri di linea sono del tutto superflui e infatti non sono richiesti. Nelle tavole 1, 2, e 3 sono indicati tutti i comandi del linguaggio. Per dare un esempio della potenza di questi comandi e della facilità d'uso dell'editor vediamo come assemblarne alcuni tra i più semplici in un piccolo programma dimostrativo.

Si seleziona il comando CLR sulla destra dello schermo e si conferma la scelta nella finestra dei messaggi. Quindi si indirizza il puntatore nell'angolo in basso a sinistra della finestra di comando e si scrolla l'elenco, quindi si seleziona il comando "scene 1 is []". A questo punto verrà caricato il catalogo degli scenari sul dischetto nel drive; quando quello chiamato jungle2 è visualizzato, basta premere il pulsante del fuoco e l'istruzione è completata. In questo modo quando il programma verrà fatto girare lo sfondo iniziale corrisponderà allo scenario

della libreria chiamato jungle2. Il passo successivo è definire un oggetto in movimento; per fare questo occorre selezionare nello stesso modo il comando "sprite 1 is []"; prima di fare questo è necessario voltare il disco Game Maker, perché la libreria degli sprite si trova sull'altro lato. Anche in questo caso viene caricato tutto il catalogo degli sprite sul dischetto.

Selezioniamo quello chiamato dog; poiché la figura del cane è composta da due sprite affiancati, comparirà automaticamente anche l'istruzione "sprite 2 is dog", cosa che sarà più chiara quando verranno esaminate le caratteristiche di SpriteMaker. Ora diamo una posizione iniziale al cane, utilizzando due istruzioni "sprite x position" e "sprite y position". Una volta selezionati i comandi e indicati con il joystick il numero dello sprite e le coordinate cartesiane (x=40 e y=154), è necessario fornire la velocità di animazione "sprite animation speed = 30", la direzione di movimento "sprite dir = 64", che corrisponde a un movimento oriz-

zontale verso destra, e la velocità di spostamento "sprite movement speed = 45".

Fatto questo non resta che verificare se funziona e se fornisce l'effetto desiderato, quindi si seleziona l'opzione RUN, sulla destra dello schermo. Se tutto è stato fatto correttamente dovrebbe vedersi un cane correre da sinistra a destra su uno sfondo di alberi. Con operazioni analoghe è possibile aggiungere oggetti in movimento, rumori e un tema musicale di accompagnamento.

• **SceneMaker.** Consente di disegnare gli scenari su cui verranno ambientati i giochi. In pratica consente di editare la mappa in alta risoluzione. Il quadro si presenta come nella figura 1. In alto a destra i comandi per la gestione dei file, per il ritorno all'editor, per l'annullamento dell'ultima operazione sul disegno e per la pulizia del video. Più sotto sulla destra la zona per la selezione dei colori, a sinistra in alto i comandi di disegno e in

basso la zona di alta risoluzione del background. Per quanto riguarda i comandi, alcuni sono i soliti: LINE disegna una linea retta tra due punti, BOX disegna un rettangolo descritto dalla posizione dell'angolo in alto a sinistra e in basso a destra, CIRCLE disegna un cerchio a partire dal centro e dal raggio, DRAW disegna a mano libera. MOVE e COPY rispettivamente spostano e copiano porzioni di disegno in altre zone, mentre FILL in aggiunta ai comandi per il disegno di aree chiuse le riempie di colore. VIEW consente di pulire lo schermo dal testo e di vedere per intero la pagina di alta risoluzione. ZOOM consente di ingrandire una porzione della scena e di editarla punto per punto.

• **SpriteMaker.** E' un potente sprite editor che oltre a offrire la possibilità di vedere l'oggetto creato sulla griglia in dimensioni reali, permette di affiancarlo ad altri sprite, facilitando la creazione di figure di grosse dimensioni e composte da un massimo di 4 sprite (che in modo espanso coprono una superficie di 96 x 84 pixel). Tuttavia gli sprite sono utilizzati soprattutto per definire figure in movimento e quindi con una certa animazione. Il cane che corre per esempio non compie solo un movimento orizzontale, ma modifica continuamente anche la posizione delle zampe e del corpo per rendere meglio l'effetto della corsa. Come probabilmente sapete l'animazione viene creata sostituendo un'immagine con una leggermente diversa fino a completamente del movimento: naturalmente è essenziale che lo scambio sia veloce e che si possa disporre di un buon numero di fotogrammi per rendere più graduale possibile il movimento. A questo scopo SpriteMaker mette a disposizione ben 32 sprite per ogni animazione e un apposito comando "anima" per testarne rapidamente l'efficacia. L'area di lavoro si presenta divisa in 4 zone (figura 3): in alto a destra la zona dei comandi, più sotto quella della selezione dei colori, a sinistra la griglia di disegno e in basso a sinistra la zona di visualizzazione degli sprite in dimensione reale. Ecco una descrizione dei comandi principali:

SETUP. Serve per dare una descrizione del tipo di sprite che si vuole disegnare: multicolor/monocromatico, espanso orizzontale, espanso verticale, numero degli sprite richiesti e posizionamento degli sprite nell'area di visualizzazione.

QUIT. Riporta all'editor.

CMD. Mostra un sottomenù di opzioni per il disegno e l'animazione: inversione degli sprite nelle quattro di-

rezioni, scorrimento dell'immagine, copia di fotogrammi (frames), pulizia e ritorno al menù.

FILE. Attiva i comandi di gestione dei file per il salvataggio, il caricamento, la cancellazione di file e l'inizializzazione del dischetto.

FR. Seleziona il fotogramma desiderato (sono disponibili 32 frames).

SPR. Seleziona lo sprite desiderato del fotogramma corrente.

• **SoundMaker.** I suoni o rumori sono una componente indispensabile dei videogame e servono a coinvolgere il giocatore nelle vicende che si realizzano sullo schermo. Esplosioni, spari, cadute, salti, laser, se sottolineati da rumori opportuni contribuiscono a rendere più reale la situazione. Il C64 anche in questo senso è veramente ben fornito, tuttavia gran parte della potenza non è disponibile da Basic e in ogni caso il lavoro di messa a punto è lungo e difficile se non si dispone di sintetizzatore. SoundMaker risolve elegantemente questi problemi fornendo quello che a prima vista potrebbe sembrare la consolle di un ingegnere del suono (figura 4). Tutto quello che è possibile fare con il SID (Sound Interface Device) viene gestito dal quadro di comando cambiando i valori delle singole voci, girando le manopole e alzando e abbassando gli indicatori. Ogni effetto può essere breve come un beep o complesso come un bombardamento; per questo è possibile, come per le immagini, disegnare le singole componenti elementari (frames) che possono poi essere linkate insieme. Ogni effetto può essere articolato su un massimo di 511 frames. Si può ascoltare il suono intero, il singolo frame, normalmente o in modo continuato, e mentre si ascolta si possono modificare tutti i valori riguardanti il frame corrente. Anche per il suono sul retro del dischetto è disponibile una ragguardevole libreria di effetti predefiniti.

• **MusicMaker.** Programmare musica non è certamente da tutti. Per farlo è necessaria non solo una notevole abilità nella programmazione, ma anche una buona conoscenza della musica e delle tecniche. Spesso i programmi per la composizione sono difficilmente abordabili da chi è alle prime armi in fatto di musica e finiscono quindi per scoraggiare. MusicMaker grazie alla sua semplicità d'uso svincola il programmatore da inutili nozioni tecniche e consente rapidamente di editare, controllare, modificare e suonare brani musicali, semplicemente ricopiando lo spartito (figura 5). Il pan-

nello di controllo sulla destra offre i comandi per il caricamento e il salvataggio dei brani, la loro editazione e correzione. E' possibile visitare lo spartito in lungo e in largo e inserire o togliere o modificare una singola nota delle tre voci disponibili. Selezionando il tempo si cambia la velocità di esecuzione, mentre dal pannello in basso a destra si selezionano gli strumenti che eseguono le parti riservate alle singole voci. Come nel caso dei rumori, anche qui i parametri possono essere variati mentre il brano suona, agevolando al massimo la sperimentazione. Nella libreria sono disponibili 17 motivi.

In conclusione, GameMaker è decisamente quanto di più potente esiste per la creazione di giochi, ma non solo: ogni singolo tool può creare oggetti che possono essere utilizzati per altri scopi. Lo stesso discorso vale per il generatore di suoni e di sprite. La novità assoluta sta nella facilità con cui si può passare da un programma all'altro e nel modo in cui si riesce ad assemblare i vari pezzi del mosaico disinteressandosi completamente della gestione della memoria, delle locazioni, dei puntatori e così via. Tutte le complicate procedure basate su ermetici PEEK e POKE sono completamente trasparenti grazie al linguaggio di sviluppo dell'editor. La memoria disponibile per la stesura del codice del gioco è di 3553 linee. Tutte le istruzioni valgono una linea, eccetto le istruzioni di print che ne valgono 6 e i commenti che contano 7 e 1/4. Gli oggetti che vengono caricati (sfondi, sprite e musiche) valgono in proporzione allo spazio occupato. Lo spazio libero viene sempre visualizzato dopo il caricamento nella parte alta dello schermo. Possono essere utilizzate un massimo di 255 label e fino a 127 strutture if then... Superato tale limite il tentativo di un'ulteriore introduzione produrrà un messaggio di errore. Il manuale, di circa 200 pagine, descrive accuratamente e in italiano il pacchetto ed esaurisce progressivamente ogni argomento senza costringere a rimandi ai vari testi in circolazione. Unica carenza: non viene descritto come vengono salvati e dove vanno memorizzati i file creati dai vari tools. Per utilizzare quindi gli oggetti creati con GameMaker al di fuori dell'editor del pacchetto occorreranno un po' di sperimentazioni.

Marco Gussoni

ABBONARSI CONVIENE...

Con RadioELETTRONICA & COMPUTER conviene ancora di più. Perché:



1

Paghi 11 numeri e ricevi a casa tua, senza aggiunta di spese postali, 12 numeri, con un risparmio di 12.000 lire.

2

Riceverai a stretto giro di posta un regalo sicuro e utilissimo. A tua scelta o un dischetto vergine di 5 pollici e 1/4, oppure un bellissimo portacassette, indispensabile per tenere in ordine i tuoi programmi.

3

Ti metti al riparo da eventuali aumenti di prezzo. Infatti, il prezzo dell'abbonamento è bloccato per tutta la sua durata.

Non perdere tempo. L'abbonamento a RadioELETTRONICA & COMPUTER per 12 numeri costa solo 60.000 lire, anziché 72.000, con l'omaggio di un dischetto vergine o di un portacassette (estero, senza dono, 80.000 lire).



SÌ! VOGLIO ABBONARMI A Radioelettronica & Computer

Cognome e nome _____

via _____

città _____

cap _____ provincia _____

nuovo abbonamento rinnovo rinnovo anticipato

Scelgo il seguente dono:

- Un dischetto vergine.
 Un portacassette con 10 compart.

Pago fin d'ora con:

- assegno non trasferibile intestato a Editronica srl
 versamento sul conto corrente postale n. 19740208, intestato a Editronica srl, corso Monforte 39, 20122 Milano (allego ricevuta)
 con la mia carta di credito BankAmericard numero scadenza autorizzando la Banca d'America e d'Italia ad addebitare l'importo sul mio conto BankAmericard

Abbonamento a 12 numeri, con dono, lire 60.000.

Abbonamento estero a 12 numeri, senza dono, lire 80.000.

Data _____

Firma _____

Spectrum 48K



JUDO

Si chiama judogi: è il costume bianco, giacca e pantaloni, che indosserete per allenarvi; e non dimenticate la cintura: bianca all'inizio e poi, via via che diventerete più bravi, di colore diverso. Non dimenticate neppure fasce e rinforzi paracolpi, però, e una buona mezz'ora di ginnastica di riscaldamento...

Il judo è una delle più nobili, diffuse e progredite arti marziali e sportive dell'Oriente. Ideato in Giappone dal grande maestro Jogoro Kano, nato nel 1860 e scomparso nel 1938, è fondato sulle basi della ancor più antica tecnica del ju-jutsu, ed è una disciplina atta a raggiungere e mantenere un perfetto sviluppo fisico, oltre che un'arte per elevare lo sviluppo mentale e soprattutto uno strumento per una efficace difesa personale.

Esiste un basilare metodo di insegnamento (studiato dallo stesso Kano), denominato go-kyo, che comprende 40 tecniche del judo, ritenute le più importanti per la loro indiscutibile potenzialità didattica e formativa. Il go-kyo si compone in particolare di 5 gruppi (detti kyo) di 8 movimenti ciascuno: il primo kyo è formato dalle otto mosse più semplici, il quinto kyo da quelle più impegnative (ed efficaci per la difesa personale).

Ciascuno dei 40 movimenti possiede la tecnica per lo studio di quello successivo, secondo un concetto didattico progressivo dal punto di vista atletico e strategico; l'attaccante mette in pratica la mossa offensiva, e il ricevente la subisce o mette in atto a sua volta quella più adatta per difendersi o per attaccare a sorpresa. Judo I è un

programma che contiene, oltre a una breve parte introduttiva che spiega le origini del judo e le caratteristiche del go-kyo, anche una vasta e completa esposizione del primo kyo e del secondo kyo (per un totale di ben 16 movimenti), ed è la parte fondamentale e propedeutica per poter studiare il programma judo II (ovvero le altre 24 tecniche proprie del terzo, quarto e quinto Kyo).

Un sistema operativo basato su videate di accesso opzionale tramite scelta diretta da parte dell'utente permette di accedere con immediatezza e semplicità alle varie sezioni previste (in totale 18). Il programma è stato scritto su Spectrum Plus, ma può funzionare perfettamente su qualsiasi Spectrum tradizionale (Issue 1, 2, 3, eccetera) purché dotato di almeno 48 Kbytes RAM: il software si spinge infatti fino alla locazione di memoria numero 63.167.

Ragioni di carattere pedagogico sconsigliano l'uso del programma a utenti di età inferiore ai 12 anni: nessun problema per quanto riguarda invece altezza, corporatura e forza fisica dei praticanti, in quanto il judo, proprio per gli scopi e i modi in cui è stato organizzato, può essere studiato e utilizzato con ottimi profitti da chiunque

si applichi seriamente (e il computer sarà un ottimo istruttore...).

Il programma

Judo I è composto da quattro file consecutivi e indipendenti (tavola 1) che sono, nell'ordine di carica: sottoprogramma di lancio (basic program), screen di copertina (bytes), generatore di grafica (bytes) e master (basic program).

Le principali caratteristiche operative delle linee dei programmi basic sono le seguenti.

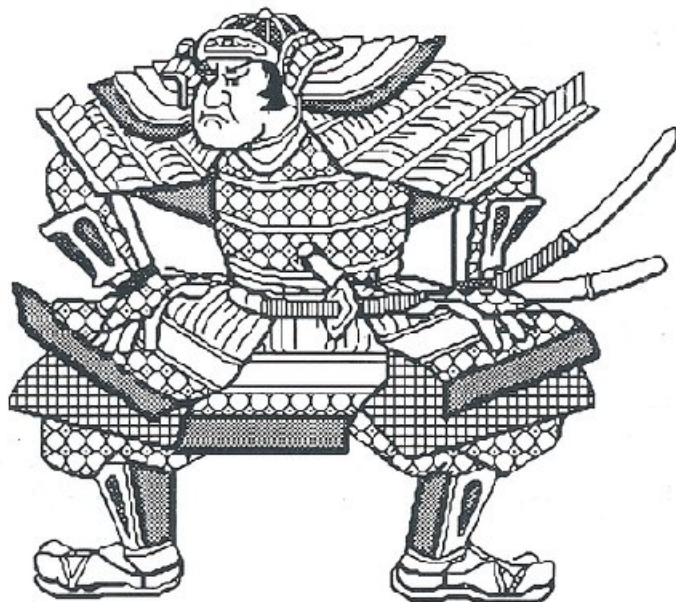
Sottoprogramma di lancio:

10-22 linee di remark; settaggio video; istruzioni per il caricamento dei tre file successivi;
23-24 data-set di copyright; stampa del titolo e del logo.

Master:

100-210 data-set di copyright; settaggio video; preparazione della videata introduttiva;
1005-1060 menù principale di accesso alle opzioni;
1110-1190 opzione 1: origini del judo;
1210-1290 opzione 2: il sistema go-kyo;

Anche tu cintura nera!





1310-1390 opzione 3: primo kyo;
 1410-1490 opzione 4: secondo kyo;
 3100-4990 data-line di definizione del testo e delle variabili di controllo;
 8010-8099 memorizzazione delle variabili principali permanenti;
 8110-8199 CLS verticale parziale del video a inizio indirizzato;
 8210-8299 stampa centrata dei testi;
 8310-8395 routine di memorizzazione dei testi da stampare;
 8401-8459 esecuzione della videoscrittura automatica indirizzata;
 9985-9989 subroutine di stampa del messaggio di avviso iniziale "FERMA IL NASTRO";
 9991-9999 data-set di copyright; spostamento del puntatore grafico dei caratteri alla locazione RAM numero 62.144.

Judo si carica in 199 secondi con un LOAD "236" oppure con il classico LOAD "".

Funzionamento

Dopo la carica del programma, parte dello screen iniziale di copertina viene cancellato (rimane solo il titolo a tutto video nella parte superiore) e compare il messaggio introduttivo "FERMA IL NASTRO", dopodiché viene proiettato il menù principale (a fondo giallo) che consente di scegliere 4 opzioni: le origini del judo (tasto 1), comprendenti una serie di elementi conoscitivi e di definizioni; il sistema go-kyo (tasto 2), in cui vengono descritte le principali caratteristiche e peculiarità tecniche del metodo; il primo kyo (tasto 3), comprendente i primi otto movimenti da imparare; infine il secondo kyo (tasto 4), composto dalle successive otto tecniche. Scegliendo queste due ultime opzioni (primo e secondo kyo) si accede ai menù di opzione secondari che consentono di selezionare le specifiche tecniche scelte

(tramite un tasto numerico da 1 a 8), che sono:

Primo kyo

1. de-ashi-barai (sgambetto al piede avanzante)
2. hiza-gumura (trattenuto ruotato al ginocchio)
3. sasae-tsuri-komi-ashi (sgambetto trattenuto alla cavaglia)
4. uki-goshi (colpo d'anca con presa in cintura)
5. o-soto-gati (sgambetto indietro al polpaccio)
6. o-goshi (grande colpo d'anca con presa alla scapola)
7. o-uchi-gari (falciata interna al polpaccio)
8. seoi-nage (braccio in spalla)

Secondo kyo

1. ko-soto-gari (piccolo sgambetto esterno al tallone)
2. ko-uchi-gari (piccolo sgambetto interno al tallone)
3. koshi-guruma (colpo d'anca con presa di testa)
4. tsurikomi-goshi (colpo d'anca raccolto)
5. okuri-ashi-barai (sgambetto laterale al mallecolo)
6. tai-otoshi (sgambetto di gamba in avanti)
7. harai-gosmi (colpo d'anca con spazzata di coscia)
8. uchi-mata (sgambetto interno di coscia)

Di ognuna di queste tecniche disponibili, a scelta avvenuta, vengono fornite la stampa del nome a centro video e la proiezione, attraverso diverse videate (sempre almeno 5), di tutto quello che è necessario sapere su:

- **Premessa**, ovvero la descrizione del tipo di movimento, della situazione che causa, del risultato che permette di conseguire a chi la mette in pratica.

- **Squilibrio**, cioè la situazione di squilibrio del corpo più appropriata.

- **Opportunità**, in pratica il momento più favorevole e la condizione dell'avversario ottimale per mettere in pratica la mossa stessa.

- **Tecnica di preparazione**, ovvero l'insieme delle procedure tecniche e strategiche che portano al compimento della mossa.

- **Movimento finale**, comprendente le posizioni e le mosse da portare a compimento per realizzare il risultato sperato.

Alla fine di ogni opzione è possibile continuare (<ENTER>) oppure tornare al menù principale (<SPACE>).

Daniele Malavasi
continua

TAVOLA 1. Caratteristiche dei file

N.	Nome	Tipologia Funzione	Istruzioni per la memorizzazione	Tempo carica
1	236C	Basic program sottoprogramma lancio	SAVE"236C"LINE1	12 sec
2	COVER	bytes screen di copertina	SAVE"COVER"SCREENS	42 sec
3	CHR	bytes generatore grafica	SAVE"CHR"CODE62144,1024	14 sec
4	236P	Basic program Master	SAVE"236P" LINE112	129 sec

Commodore 64



WAR GAME

Guerra Napoleonica è un gioco strategico per due persone, che si ispira alle battaglie del 1802; da buon war game, tiene conto di molte variabili per stabilire via via vittorie e sconfitte. I due giocatori, che comandano le forze napoleoniche e quelle alleate, dispongono di 10 unità ciascuno e devono conquistare il territorio del nemico e raggiungere il punteggio necessario per vincere.

Al Commodore l'ardua sentenza

Vi ricordate i giochi di simulazione bellica? Qualche anno fa ebbero la loro stagione di successo, con tanto di club e tornei. Erano diversi da ogni altro gioco da tavolo, perché richiedevano ore e ore per portare a conclusione una partita, anzi, una battaglia. Ogni mossa richiedeva una lunga meditazione strategica: il morale delle truppe, la pendenza del terreno, la gittata dei cannoni, la tattica del nemico erano variabili da ben considerare volta per volta, da soppesare attentamente prima di sferrare un attacco, prima di impegnare un contingente in una manovra di accerchiamento.

Tanto duravano quelle battaglie, che tutte le mamme brontolavano e si disperavano: il tavolo più grande della casa era impegnato per intere settimane dalla mappa enorme del territorio, sulla quale pedine, soldatini, segnali, dadini, simboli e bandiere stazionavano intoccabili. Guai a spostarli per spolverare! Come ci si poteva ricordare, se no, l'ordine di battaglia? La disposizione delle armate? I morti e i feriti?

Per rivivere il fascino di quei combattimenti, senza rinnovare però anche quelli parentali, eccone una versione per computer: Guerra Napoleonica è un gioco strategico per due persone che si ispira alle battaglie del 1802. I due giocatori sono al comando delle forze napoleoniche (in reverse) e delle forze alleate e dispongono di 10 unità ciascuno, 3 flotte e 7 armate, per sconfiggere l'avversario. Lo scopo del gio-

co è conquistare territorio nemico fino a raggiungere il punteggio necessario per vincere.

Come si gioca

All'inizio del gioco viene mostrato il punteggio base delle forze napoleoniche e il punteggio che i due schieramenti dovranno totalizzare per vincere. Le forze napoleoniche vincono se raggiungono la quota di 50 punti, le forze alleate se mandano a zero il punteggio di quelle napoleoniche. Nelle figure 1 e 2 sono visibili le schermate con l'elenco delle unità a disposizione delle due forze e le relative disponibilità di uomini, numero di identificazione e base di partenza. È molto importante conoscere bene la base di partenza delle proprie unità poiché se durante il gioco cadono al nemico escono dall'azione. Nella schermata successiva compare la mappa di gioco (figura 3). Il territorio degli alleati appare in modo normale, mentre quello napoleonico in reverse. Premendo un tasto, sulla mappa vengono visualizzate le unità (flotte e armate) a disposizione. Le unità sono riconoscibili dal numero di identificazione (i numeri da 0 a 2 rappresentano le flotte mentre i numeri da 3 a 9 le armate).

Una volta che i due giocatori hanno osservato la propria situazione si passa alla fase di comando. Durante questa fase i due giocatori impartiscono gli ordini, separatamente e in segreto, alle proprie unità, indicando la direzione

di movimento. Inizia il giocatore che comanda le forze alleate; quindi, dopo avere impartito gli ordini a tutte le unità a sua disposizione, tocca a quello che comanda le forze napoleoniche. Il nome di ogni unità viene visualizzato in alto a sinistra sullo schermo insieme alla relativa disponibilità di uomini (F) e al livello di efficienza e di morale (M&E).

In alto a destra sono visualizzati i numeri da 1 a 8 per indicare gli 8 movimenti che ogni unità può compiere durante ogni anno di guerra. Per dare gli ordini di movimento basta digitare uno dei seguenti caratteri:

- "N" per andare a nord
- "S" per andare a sud
- "O" per andare a ovest
- "E" per andare a est
- "*" per rimanere fermi

Durante l'immissione degli ordini è possibile correggere eventuali errori usando il tasto ed è anche possibile uscire dal programma premendo F1 (il computer in questo caso chiede conferma). Per passare all'unità successiva è necessario introdurre 8 comandi di movimento e quindi premere <return>. Una volta premuto <return>, però, non è più possibile modificare gli ordini dati.

Movimento

Le flotte possono muoversi ovunque nel mare e possono occupare an-

UNITA'	NUM.	UOMINI	BASE
ARMATA	0	6000	BREST
ARMATA	1	5000	TOURNE
ARMATA	2	8000	CADICE
ARMATA	3	40000	PARIGI
ARMATA	4	40000	PARIGI
ARMATA	5	40000	PARIGI
ARMATA	6	40000	PARIGI
ARMATA	7	40000	PARIGI
ARMATA	8	40000	BORDEAUX
ARMATA	9	35000	LONDRA

FORZE NAPOLEONICHE

Figura 1. Forze dello schieramento Napoleonico.



Figura 2. Mappa dell'Europa divisa nei territori dei due schieramenti.

che le zone costiere, sia amiche sia nemiche. Se tuttavia hanno occupato un quadrato di terra, possono muoversi successivamente solo su un quadrato di mare (figura 4).

Le armate possono muoversi su ogni quadrato di terra, sia amico sia nemico, ma non possono muoversi sui quadrati di mare. Nessuna unità può muoversi oltre i limiti del terreno di gioco (quindi non è possibile fare uscire una armata da un lato dello schermo per farla ricomparire dall'altro).

I caratteri del seme di quadri (i rombi della figura 1) rappresentano terreno difficoltoso (montagne, fiumi, paludi) e c'è il 50% di probabilità che fallisca un tentativo di attraversarli. I quadrati che contengono una lettera rappresentano le città e, se appartengono al nemico, anche per essi c'è il 50% di probabilità che fallisca un tentativo di attraversamento. Le linee di confine, i punti (figura 4) e il territorio nemico non danno, invece, alcun problema al movimento delle unità.

Ogni volta che viene attraversato del territorio nemico (comprese le città), il colore del quadrato corrispondente cambia, cioè passa in reverse se una unità napoleonica attraversa un quadrato di terreno confederato e passa da reverse a normale se una unità alleata attraversa un quadrato di terreno napoleonico, e il punteggio viene incrementato di 1 (poiché viene segnato solo il punteggio dei francesi, se gli alleati conquistano un quadrato di terreno il punteggio delle forze napoleoniche verrà decrementato di 1). I caratteri del seme di quadri non cambiano colore e non danno alcun contributo al punteggio se vengono attraversati. Se una unità non può effettuare il movimento ordinato resta ferma e quindi e-

segue, se possibile, solo gli ordini successivi. Nessuna unità può muoversi su un quadrato occupato da una unità dello stesso schieramento.

Fa eccezione l'imbarco: una armata può essere imbarcata su una flotta del proprio schieramento e quindi trasportata attraverso il mare. Una volta in mare, però, è necessario che l'armata segua il movimento della flotta altrimenti verrà persa. L'armata può essere imbarcata solo se la flotta viene posizionata nel quadrato di mare contiguo alla costa.

Risultati

Dopo che entrambe le parti hanno dato i loro ordini in segreto, sullo schermo compaiono i risultati relativi alle otto mosse per l'anno di guerra in corso (gli otto ordini dati a ciascuna unità equivalgono a un anno di guerra). Si muovono per prime le flotte, quindi le armate, entrambe in ordine casuale. Se una unità si muove su un quadrato occupato da una unità nemica allora ha luogo una battaglia.

Se l'unità in difesa (cioè quella che si trova già sul quadrato) si trova in una città amica (una città del proprio schieramento oppure una città nemica attraversata) il suo potenziale viene moltiplicato di un fattore uguale a 5 (il potenziale delle unità è funzione del grado di efficienza e morale, della quantità di uomini e di un coefficiente di casualità). Questo evento viene segnalato dal carattere "*" a fianco dell'unità che gode del vantaggio. Sul video compaiono anche le perdite di uomini oltre all'esito finale della battaglia. Se la forza in difesa vince, tutti i rimanenti ordini dell'unità attaccante vengono cancellati. Questo serve per evitare che una unità attacchi ripetuta-

mente sino alla distruzione.

Se l'unità che attacca vince, quella in difesa è costretta a indietreggiare nella direzione dell'attacco e l'unità che ha attaccato occupa la posizione lasciata libera. Se l'unità in difesa che ha perso indietreggia sul territorio nemico, fuori dai limiti della mappa o, nel caso delle armate, in mare, viene automaticamente eliminata. Se una flotta distrutta trasportava una armata questa non viene persa nello scontro a patto che successivamente la sua posizione rimanga immutata (in pratica è come se fosse ancora imbarcata). In questo caso l'armata dovrà rimanere ferma fino a quando potrà essere recuperata con una flotta.

Il morale e l'efficienza delle unità cambiano con il risultato delle battaglie. Quindi perdendo le battaglie questo fattore diminuisce, mentre cresce se si vincono. Se il grado di efficienza e morale di una unità va a 0 l'unità viene rimossa. Oltre alla vittoria e alla sconfitta, l'altro risultato possibile di una battaglia è l'esito incerto. In questo caso vengono solo calcolate le perdite e quindi sottratte alle rispettive forze.

Dopo che sono stati mostrati gli effetti degli ordini dati alle unità durante l'anno di guerra in corso, il gioco riprende dalla fase di comando e i due giocatori devono dare i nuovi ordini per l'anno seguente. Al termine di ogni anno di guerra le forze rimaste di ciascuna unità vengono incrementate di un numero di uomini proporzionale alla quantità preesistente e al livello di morale e di efficienza moltiplicato per un valore pseudocasuale.

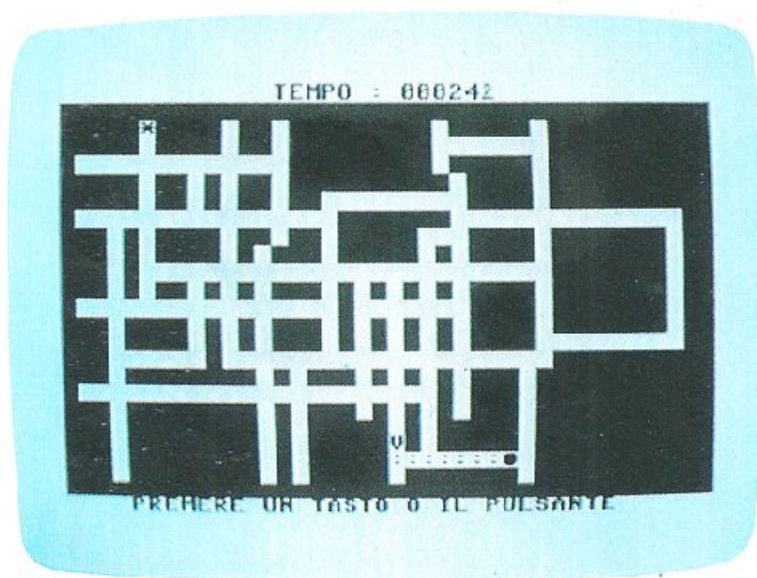
A fine partita viene chiesto se si vuole giocare un'altra partita. Digitando "N" si esce dal programma.

Paolo Gussoni

Commodore 64



GIOCHI



Senza Arianna si fa così

Per viverlo davvero, bisogna viverlo dal di dentro, il dedalo: sullo schermo compare via via un'immagine prospettica che mostra il punto in cui siete, esattamente come la vedreste in un labirinto vero. Labirinth costruisce per voi migliaia di percorsi, con undici livelli di difficoltà.

Dedalo! Secoli di storia e di fantasia nulla hanno tolto, e nulla hanno saputo aggiungere, a un fascino dei più antichi, dei più paurosi ed eccitanti: perdersi e non trovare la via del ritorno in un intrico uguale e diverso di svolte e corridoi, nel quale metro dopo metro diminuisce la speranza e aumenta la confusione. Perché non sempre c'è la buona Arianna con il suo magico filo...

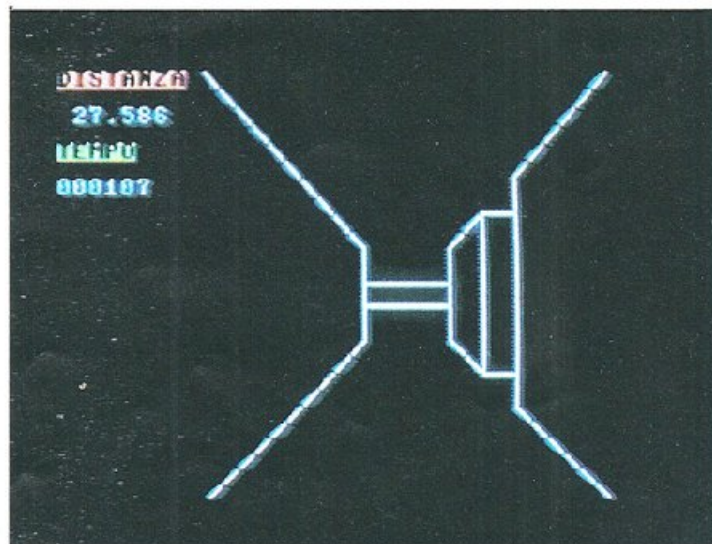
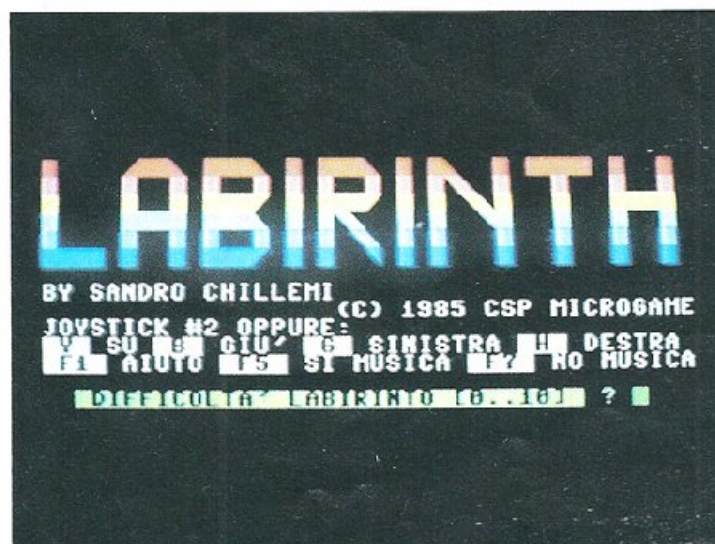
Nei giochi di carta, e anche in alcune versioni per computer, i labirinti vengono presentati sempre dall'alto: voi ne siete fuori, superiori ed estranei, e quello che c'è dentro, e che ne deve uscire, è la punta della vostra matita, oppure il cursore del vostro computer. Nulla a che vedere con la

sensazione di sgomento che si prova dal di dentro, quando bisogna decidere, di fronte a una parete di pietra, se girare a destra o a sinistra...

Labirinth vi consente di goderla.

Come si gioca

Una volta avviato il programma, compare in caratteri multicolori il nome Labirinth, accompagnato dalle istruzioni, chiare e telegrafiche. Va selezionato subito il grado di difficoltà, da 0 a 10, del labirinto che si vuole affrontare, e appare la scritta "Sto preparando un nuovo labirinto"; durante l'attesa strisce colorate che colmano lo schermo segnalano il tempo che passa (pochi secondi).



Io programmo, tu progetti, egli pianifica...

La pubblicità su Radio Elettronica & Computer è informazione. Chi legge Radio Elettronica & Computer possiede un personal. E vuol sapere che cosa gli permetterà di sfruttare meglio il suo computer. Una nuova stampante? Un nuovo programma? Un nuovo modem? Una nuova tavoletta grafica? Una nuova banca dati? Una nuova marca di dischetti? Per chi legge Radio Elettronica & Computer è importante saperlo. E subito.

Per la pubblicità su Radio Elettronica & Computer, potete rivolgervi a STUDIOSFERA (Sig. Berardo - Sig. Fracassi) tel. 02/7533939 - 7532151

Senza Arianna si fa così

Inizia il gioco: un pallino rosa, che siete voi, si trova di fronte a un muro. Può provare a girare a sinistra: intravede così un corridoio. A mano a mano che avanza in linea retta, si aprono sia a destra sia a sinistra dei corridoi laterali. Quale scegliere?

Meglio premere subito il tasto F1, cioè il tasto di aiuto, e dare uno sguardo dall'alto alla mappa del labirinto, che indica la vostra posizione e quella dell'obiettivo da raggiungere, che non è l'uscita, ma un qualche tesoro da conquistare. Bravo è, naturalmente, chi a questo tasto ricorre solo all'inizio, e si imprime nella memoria il percorso a cui tener fede. Ma non è facile.

Bravo è inoltre chi raggiunge l'obiettivo nel minor tempo possibile, e quindi con un percorso "pulito", non troppo pasticciato da ripensamenti. Per valutare la propria abilità basta tener d'occhio il timer, che parte da zero all'inizio del gioco e conta i secondi. Un altro contatore, utilissimo per orizzontarsi, segna la distanza in metri (con tre decimali!).

Non è indispensabile avere il joystick (da collegarsi alla porta 2): per decidere le tre direzioni possibili e l'eventuale dietrofront basta usare i tasti centrali: Y va in avanti, G gira a sinistra, H gira a destra e B inverte la direzione di marcia. Altri tasti da usare sono F5 e F7, che rispettivamente attivano e disattivano l'allegria colonna sonora.

Quando si giunge alla meta compare un messaggio di benvenuto e, con la pressione di return, la veduta dall'alto del percorso effettuato. Premendo un tasto si torna all'inizio e si può scegliere un altro percorso.

Tre note tecniche

Nella ricostruzione dello schermo, Labirint è molto più veloce degli altri programmi simili in circolazione, perché, mentre il corpo principale è in Basic compilato con il pet speed, le routine che richiedono maggiore velocità, come per esempio l'effetto di scrolling quando si va a sbattere contro il muro, sono in linguaggio macchina. L'algoritmo che genera il labirinto è molto sofisticato. Infatti a differenza di molti giochi analoghi è impossibile restare chiusi, senza almeno una strada possibile, anche quando il livello di difficoltà impone il massimo numero di vicoli ciechi. Una volta lanciato il programma, non è più possibile arrestarlo: sono stati disabilitati infatti i tasti STOP e STOP/RESTORE. Quindi per salvarlo è necessario operare prima di dare il RUN.

Sandro Chillemi

**Spectrum
16K**

GRAFICA

Grazie alla trigonometria è possibile tracciare sul video solidi tridimensionali partendo da due punti di fuoco e un orizzonte. Il risultato è di ottimo livello, e imparare a usare il programma non è poi così difficile...

Lo spigolo, il fuoco, la terra

Questo programma si avvale di formule trigonometriche per la determinazione di solidi tridimensionali in uno spazio dove vi sono due punti di fuoco e una linea di terra, orizzonte. Entra perfettamente nei 16 K di uno ZX Spectrum, anche se è notevolmente rallentato dalla presenza di moltissime conversioni di alcune stringhe contenenti caratteri numerici che servono appunto per mantenere il programma entro i 16 K. Quindi chi possiede uno Spectrum con espansione potrà sostituire ogni conversione da stringa a numero con un valore numerico, aumentando notevolmente la velocità di calcolo delle coordinate spaziali e del movimento del cursore con le solite frecce. I solidi che vengono costruiti sullo schermo sono solo parallelepipedi, come quello che appare, per esempio, in figura 1.

Come funziona

Lo spigolo in primo piano del solido può essere oltre la linea di terra, oppure intersecare la medesima, però non può essere tra il margine video e un punto di fuoco: in questo caso il computer, oltre a non accettare le coordinate, si fermerebbe riportando errore. Le formule applicate ai triangoli rettangoli per stabilire con esattezza le coordinate sono visibili nella figura 2. Ogni vertice del solido possiede una propria variabile, anzi più precisamente ne possiede due: coordinata verticale e coordinata orizzontale. Queste variabili sono articolate così: quando la variabile ha nel proprio testo il numero 1 significa che è una coordinata orizzontale; per esempio la variabile A1 conterrà il valore orizzontale, mentre quello verticale è contenuto nella

variabile A. Naturalmente vi sono altre variabili che vengono usate come flag; per esempio quando la variabile SN è a 1 significa che il solido ha lo spigolo oltre la linea di terra, infatti il computer dovrà eseguire particolari calcoli per individuare le coordinate, sempre con le medesime routine, ma illudendo il programma. Gli si fa credere cioè che il solido sia sotto la linea di terra cosicché viene costruito, solo nelle variabili, un solido simmetrico rispetto alla linea di terra; pertanto il flag sopra menzionato servirà a ricordare al computer se il solido è simmetrico e quindi se stamparlo sopra la linea di terra oppure no. La figura 3 vi chiarirà le idee a questo

proposito. In un primo tempo il programma calcolava la profondità punto per punto, perché non esiste una vera formula; con questo sistema però il calcolo della profondità diventava assai lungo, specialmente quando il vertice basso e in primo piano si trovava vicinissimo alla linea di terra. In seguito, eguagliando una relazione trigonometrica, si è ricavata una formula per accelerare i calcoli: il tempo medio si è abbassato da 3 a 4 minuti a circa 5 secondi. La formula è visibile nella figura 4.

Durante la costruzione del solido, il computer vi pone alcune domande che servono per decidere se stampare o no un lato del parallelepipedo. Nella fi-

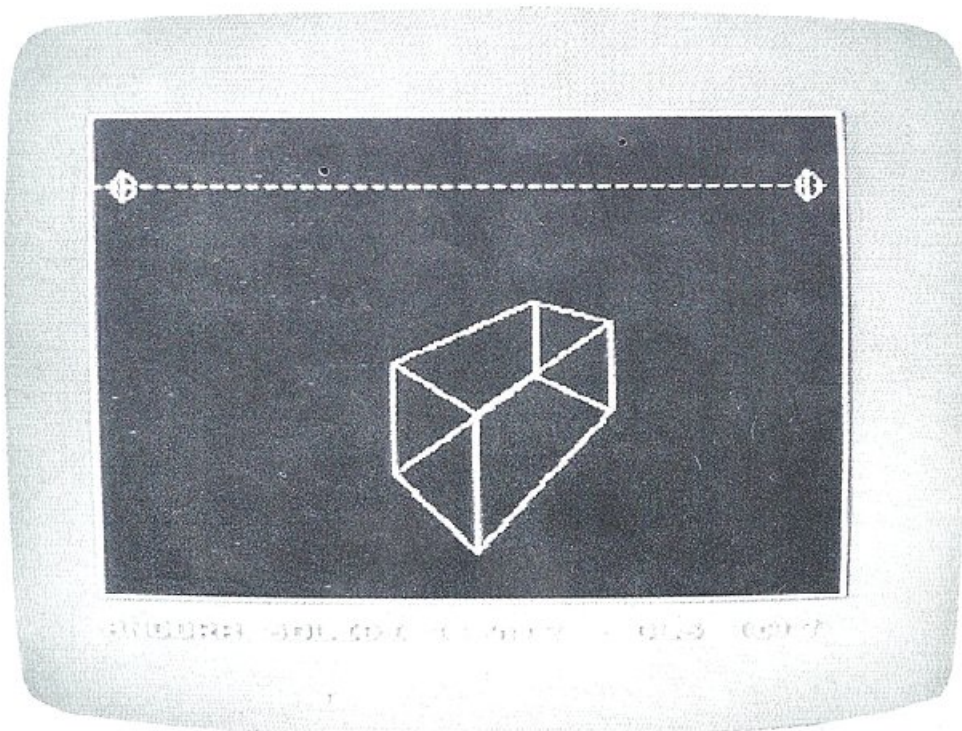


Figura 1. Esempio di parallelepipedo costruito con il programma 3D.

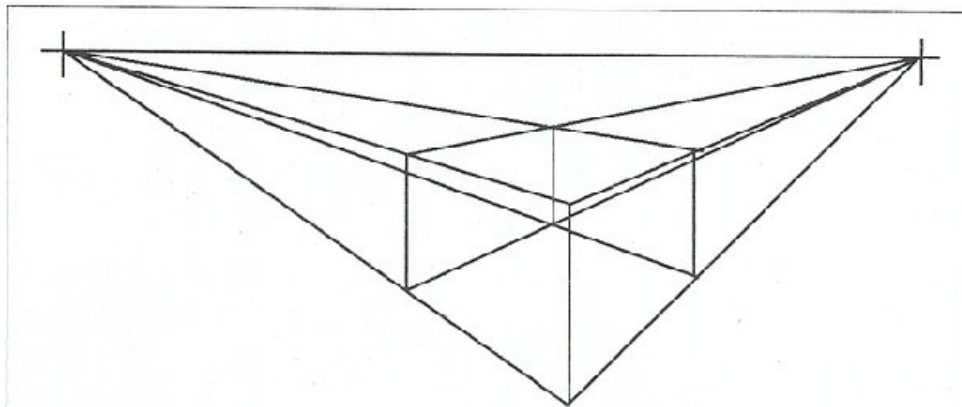


Figura 1. I solidi tridimensionali realizzati dal programma sono parallelepipedi.

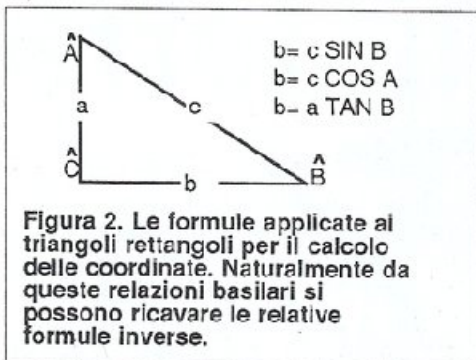


Figura 2. Le formule applicate ai triangoli rettangoli per il calcolo delle coordinate. Naturalmente da queste relazioni basilari si possono ricavare le relative formule inverse.

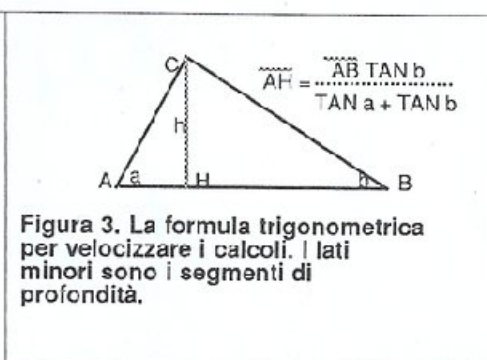


Figura 3. La formula trigonometrica per velocizzare i calcoli. I lati minori sono i segmenti di profondità.

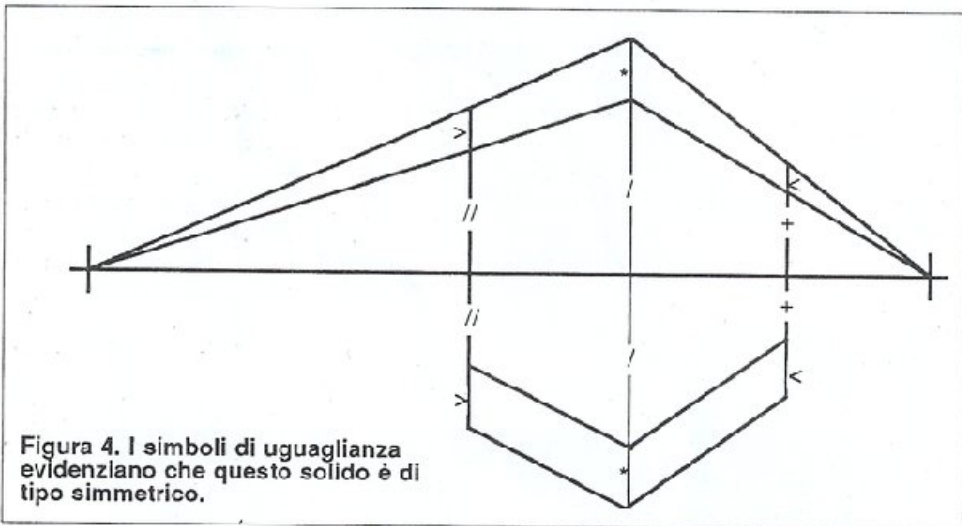


Figura 4. I simboli di uguaglianza evidenziano che questo solido è di tipo simmetrico.

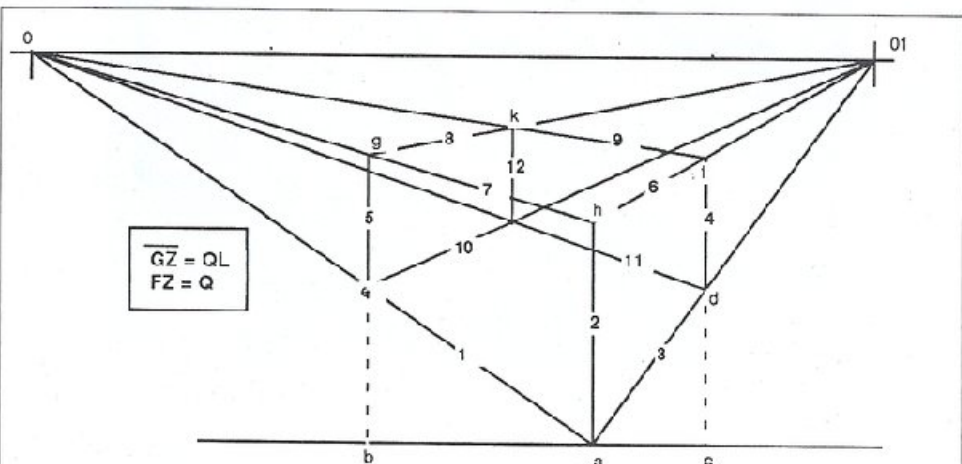


Figura 5. Questo solido è correlato dalle variabili a ogni suo vertice, a ogni lato e ad alcune proiezioni di lati su altri lati.

Lato	Relativa domanda
1	stampa lato sx (1') ?
2	stampa altezza ?
3	stampa lato dx (1') ?
4	stampa spig. dx ?
5	stampa spig. sx ?
6	stampa lato dx sup. (1') ?
7	stampa lato sx sup. (1') ?
8	stampa lato sx sup. (2') ?
9	stampa lato dx sup. (2') ?
10	stampa lato sx (2') ?
11	stampa lato dx (2') ?
12	stampa spig. (2') ?

Figura 5 ogni lato è numerato in modo da corrispondere alle relative domande del computer sulla stampa del lato, come vengono presentate nella tavola 1. Per ricercare a proprio piacere il punto preciso dello spigolo in basso si usa il cursore con i tasti usati per l'EDIT. Quando la coordinata è stata scelta, si preme ENTER e il programma prosegue. Se si vuole velocizzare il cursore si deve premere P, dopo di che il computer chiederà il nuovo passo, cioè il nuovo numero d'incremento che inizialmente è sempre 1. Il cursore assume sempre un colore inverso da quello delle carta, cosicché si possono controllare più facilmente i suoi spostamenti. Anche nella routine per la determinazione del lato trasversale con i tasti 5 e 8 esiste l'opzione dell'incremento del passo.

Quando si determina l'altezza si usa sempre la routine del cursore anche se viene controllata la coordinata ricavata con quella dello spigolo in basso; infatti non si può avere un'altezza negativa, cioè la coordinata dell'altezza deve essere sempre maggiore di quella dello spigolo in basso, se no viene ripetuto il rilevamento della medesima.

Altre opzioni

Alla fine della costruzione del solido si può scegliere se costruirne altri oppure no o anche se cancellare il video con il tasto C. Invece con il tasto A si abilita un'altra routine, che cancella alcune linee; il cursore è sempre mosso dai tasti dell'Edit, mentre il tasto O abilita la cancellatura con PEN-DOWN e il tasto I la disabilita con PENUD. Anche in questa routine esiste l'opzione del nuovo passo e per uscire dalla medesima si deve premere ENTER. Sempre alla fine della costruzione con il tasto D si può salvare lo schermo dando un titolo che verrà stampato in alto a sinistra.

Cristiano Frigeni

Commodore 64



UTILITY

Ecco a voi un programma che è, in pratica, un mini corso per la gestione dei file ad accesso casuale: relativi, che permettono di operare sui dati di un file senza dover tener conto della traccia e del settore su cui si sta lavorando, e random.



Gestire l'accesso casuale

L'unità floppy Commodore possiede due metodi per l'accesso casuale: i file relativi e i file casuali. I file casuali sono costituiti da singoli blocchi di 256 byte di dati. In pratica ogni blocco corrisponde a un settore di una traccia. Il dischetto formattato è diviso in tracce che hanno la forma di cerchi concentrici. Su ogni disco vi sono tracce numerate in modo che la 1 è quella più esterna e la 35 è quella più vicina al centro. Ogni traccia è a sua volta suddivisa in settori cioè appunto in blocchi di 256 Bytes. Poiché le tracce più esterne sono più lunghe, contengono un numero maggiore di settori. Le tracce esterne hanno infatti 21 settori, mentre quelle interne, solo 17. Il D.O.S. (sistema operativo del drive) contiene i comandi per accedere direttamente a qualunque traccia e settore del floppy. Questi comandi sono inviati per mezzo del canale di comando 15. Per l'apertura di un file ad accesso casuale è necessario disporre di due canali, uno per i comandi (canale 15) e uno per i dati. Il formato per le istruzioni necessarie è il seguente:

```
10 OPEN 15,8,15  
20 OPEN 5,8,5 "#"
```

Abbiamo visto quindi che il canale dati per il file ad accesso casuale viene

aperto (OPEN) ponendo come nome del file il simbolo (#).

A questo punto dobbiamo poter specificare al D.O.S. su quale traccia e settore scrivere. L'istruzione da usare in questo caso è Block-Write (blocco su cui scrivere).

Il comando B-W permette, dopo aver caricato in un buffer delle informazioni, di scrivere sul floppy in una traccia e in un settore specificato. Il formato del comando B-W nel nostro caso è il seguente:

```
PRINT #15,"B-W:"5;0;T;S
```

dove (15) è il numero del file con cui abbiamo aperto il canale, (5) è il numero del canale attraverso cui passeranno i dati, (0) è il numero del drive, T ed S rispettivamente la traccia e il settore su cui scrivere. Al nostro programma aggiungiamo ora le seguenti istruzioni:

```
10 OPEN 15,8,15  
20 OPEN 5,8,5,"#"  
30 PRINT#5,"PROVA"  
40 PRINT#15,"B-W:"5;0;1;2  
50 CLOSE15:CLOSE5
```

In questo modo abbiamo scritto nella traccia 1 settore 2 la parola "PROVA". L'istruzione che ci permetterà

l'operazione inversa a quella appena descritta è la BLOCK-READ (blocco da cui leggere).

L'istruzione B-R è praticamente analoga alla precedente:

```
PRINT#15,"B-R:"5;0;T;S
```

Aggiungiamo ora al programma delle altre istruzioni:

```
10  
-(IDENTICHE ALLE PRECEDENTI)  
50  
60 PRINT CHR$(147)"PREMERE  
UN TASTO"  
70 GETA$:IFAS="" THEN 70  
80 OPEN 15,8,15:OPEN 5,8,5"#"  
90 PRINT#15,"B-R:"5;0;1;2  
100 INPUT#5,B$  
110 PRINT "AVEVI REGISTRATO-  
":B$  
120 PRINT "FINITO"  
130 CLOSE15:CLOSE5
```

Il passo successivo ci porterà a poter scegliere quale byte modificare o leggere tra i 256 byte contenuti in un settore.

Il comando con cui è possibile eseguire questa operazione è il buffer-pointer (puntatore al buffer). Si può accedere in modo casuale ai singoli


```

10 OPEN1,8,15
20 OPEN2,8,3,"0:NOME,L,"+CHR$(42)
30 FORI=1TO41
40 PRINT#1,"P" CHR$(3) CHR$(10) CHR$(0) CHR$(I)
50 PRINT#2,"J"
60 NEXTI
70 FORN=1TO10
80 INPUT"NOME CANTANTE";N$
90 PRINT#1,"P" CHR$(3) CHR$(N) CHR$(0) CHR$(1)
100 PRINT#2,N$
110 INPUT"TITOLO DELLA CANZONE";T$
120 PRINT#1,"P" CHR$(3) CHR$(N) CHR$(0) CHR$(22)
130 PRINT#2,T$
140 NEXTN
150 CLOSE1:CLOSE2
160 PRINT"PREMERE UN TASTO PER LEGGERE"
170 GETP$:IFP$=""THEN170
180 OPEN1,8,15
190 OPEN2,8,3,"0:NOME"
200 FORN=1TO10
210 PRINT#1,"P" CHR$(3) CHR$(N) CHR$(0) CHR$(1)
220 INPUT#2,N$
230 PRINT"NOME CANTANTE-";N$
240 PRINT#1,"P" CHR$(3) CHR$(N) CHR$(0) CHR$(22)
250 INPUT#2,T$
260 PRINT"TITOLO DELLA CANZONE-";T$
270 PRINT"PREMERE UN TASTO PER CONTINUARE"
280 GETP$:IFP$=""THEN280
290 NEXTN
300 PRINT"FINITO"
310 CLOSE1:CLOSE2

```

Figura 1. Ecco il listato completo delle routine di scrittura e lettura in un file relativo.

byte all'interno di un blocco. Il formato del comando B-P è il seguente:

```
PRINT#15,"B-P:" 5;N
```

dove (15) è il numero del file con cui si è aperto il canale 15, (5) è il nume-

ro del canale aperto per accesso casuale, N è il numero del byte su cui si vuole intervenire. A questo punto scriviamo un nuovo programma:

```
10 OPEN 15,8,15
20 OPEN 5,8,5"#"
```

```
30 PRINT#15,"B-P:"5;20
40 PRINT#5,"A"
50 PRINT#15,"B-W:"5;0;1;2
60 CLOSE15:CLOSE5
```

Con questo programma si aprono ancora una volta i canali 15 e 5, si punta al byte 20, si carica il buffer con una "A" (PRINT#5,"A") e la si scrive sulla T=1 e S=2. Modificando il programma, potremo facilmente rileggere il ventesimo byte del secondo settore della prima traccia.

```

100 OPEN15,8,15
110 OPEN5,8,5,"#"
120 FOR T=1TO35
130 PRINT"TRACCE TRACCIA ";T;
PRINT" SETTORE 0":PRINT#15,"B-R:" 5;0;T;0
140 INPUT#15,A,B$,C,D
150 IFAC20 THEN180
160 PRINTA,B$,C,D
170 GETP$:IFP$=""THEN170
180 NEXTT
190 CLOSE5:CLOSE15

```

```

10 OPEN 15,8,15
20 OPEN 5,8,5"#
30 PRINT#15,"B-R:"5;0;1;2
40 PRINT#15,"B-P:"5;20
50 GET35,AS
60 PRINT "AVEVI SCRITTO UNA-";AS
70 CLOSE15:CLOSE 5

```

Figura 2. Questa routine controlla gli errori nel blocco 0 di ciascuna delle 35 tracce.

Per concludere, ricordate che quando si deve scrivere un file è necessario prima puntare al byte, quindi caricare nel buffer i dati che si vogliono scrive-

re, e per ultimo indicare la T e la S su cui scrivere. Se si deve leggere un file prima si punta la T e S su cui si vuole leggere, quindi il byte; poi lo si legge con un INPUT# o un GET#.

File relativi

Un notevole passo avanti nel trattamento dei dati ad accesso casuale lo si ottiene usando i file relativi. Questi permettono di operare sui dati che si desidera nell'ambito di un file senza dover tener conto su che traccia e settore si sta lavorando. Inoltre i file relativi permettono di suddividere i file in record e in campi all'interno del record consentendo quindi una grande elasticità nelle operazioni.

Per ben capire questi concetti è bene chiarire che cosa si intende per record e per campi in un record. Un record deve essere considerato come un insieme di byte in cui sono contenuti tutti i dati che ci interessano. Per esempio in (nome di un cantante-titolo canzone) tutto quello che si trova tra parentesi è ciò che ogni record conterrà. Supponiamo inoltre che il nome e il titolo della canzone non occupino nell'insieme più di 42 bytes. Il dato (42) rappresenta quindi in particolare quale sarà la lunghezza massima di ogni record. In generale la lunghezza massima possibile di un record è di 254 bytes. Il formato per l'apertura di un record è il seguente:

```
10 OPEN 1,8,15
20 OPEN 2,8,3,"O:NOME,L, "+
  CHR*(N)
```

Anche in questo caso è necessario aprire il canale (15) attraverso cui mandare i comandi al D.O.S e un file in cui inviare i dati veri e propri (2=n° del file, 8=n° del canale, "O:NOME,L," = forma del comando per aprire e creare un file relativo, +CHR*(N)= lunghezza massima di ogni record (nel nostro caso N=42). Se invece si vuole aprire un file relativo che è già stato creato, basta usare il seguente formato:

```
OPEN 2,8,3, "O:NOME"
```

Ora dobbiamo chiederci di quanti record vogliamo disporre nel file relativo appena aperto.

Supponiamo di disporre al massimo di 10 canzoni; si forzerà quindi il D.O.S a creare 10 record scrivendo nel decimo:

```
30 FORI=1TO41
40 PRINT#1, "P"CHR$(3),
  CHR$(10), CHR$(0), CHR$(I)
```

```
1 PRINT "ZLEGGE #DDIRECTORI O=B0AM"
2 GETW$: IFW$=" " THEN 2
3 IFW$="D" THEN 6
4 IFW$="B" THEN 7
5 IFW$<>"D" THEN 2
6 S=1:GOTO10
7 S=0
10 OPEN15,8,15
20 OPENS,8,5,"#"
25 PRINT#15,"B-R:" S;0;18;S
30 FORI=0TO255
40 PRINT#15,"B-P:" S;I
50 GET#5,A$: IFA$=" " THENA$=CHR$(0)
60 PRINT ASC(A$),I
61 IF ASC(A$)>20 THEN 100
62 NEXTI
70 CLOSE5:CLOSE15
80 END
100 PRINT "XXXXXXXXXXXX";A$;"J"
110 GOTO62
```

Figura 3. Questa routine visualizza i codici ASCII della Directory e della BAM (T=18 S=0 e T=18 S=1).

```
50 PRINT#2,"J";
60 NEXTI
```

Il formato della riga 40 introduce il comando "P"=position. Questo specifica il canale CHR\$(3) del file relativo, il numero del record su cui scrivere (CHR\$(10) CHR\$(0)), nel solito formato CHR\$(0)*256+CHR\$(10) e CHR\$(1), la posizione all'interno del record da 1 a 41 in cui scrivere una J. Per capire bene come usare un file relativo, bisogna comprendere il concetto di campo. Il record conterrà il nome del cantante e il nome della canzone per un totale di 42 byte: in pratica questi rappresentano proprio i due campi in cui abbiamo suddiviso il record.

```
NOME 20
TITOLO 20
```

20 e 20 indicano invece il numero massimo di caratteri disponibili per ciascun campo. Ponendo il tutto in ordine sequenziale vedremo che il nome inizierà al 1° byte e finirà al 20° byte. Ci sarà quindi un carattere vuoto per spaziare e al ventiduesimo carattere inizierà il titolo della canzone, che disponendo al massimo di 20 caratteri, finirà al quarantunesimo carattere (byte), anch'esso seguito da uno spa-

zio nel quarantaduesimo byte. In tutto quindi 42 byte come la lunghezza del record. Il programma per scrivere tutti e 10 i titoli e cantanti sarà per esempio:

```
70 FORN=1TO10
80 INPUT "NOME CANTANTE";
  N$
90 PRINT#1,"P" CHR$(3)
  CHR$(N) CHR$(0) CHR$(1)
100 PRINT#2,N$
110 INPUT "TITOLO DELLA CAN-
  ZONE";T$
120 PRINT#1, "P"CHR$(3)
  CHR$(N) CHR$(0) CHR$(22)
130 PRINT#2,T$
140 NEXTN
150 CLOSE 1:CLOSE2
```

Abbiamo così inserito 10 nomi e 10 titoli nei nostri record. A questo punto conosciamo tutti i "segreti" per poter concludere il programma con una routine che riapra il file relativo e rilegga tutti i 10 record. Provateci! Comunque, niente paura: se ci fossero dei problemi, potrete verificare i vostri sforzi confrontando il vostro programma con il listato di figura 1. Nelle figure 2 e 3, invece, potete vedere due esempi esplicativi.

Buon divertimento!

Luigi Brescia

Commodore 64



PROGRAMMAZIONE

Eccoci alla terza puntata del corso di programmazione in linguaggio macchina. E' di scena il modo di indirizzamento, un attributo che specifica come l'istruzione dovrà trattare il proprio operando e influisce sul codice dell'istruzione stessa.

L'operando va trattato bene

La scorsa volta abbiamo visto come è possibile compiere una semplice elaborazione aritmetica con il 6510 e manipolare alcuni dei flag del registro di stato. Nell'esempio venivano addizionati due numeri che potevano essere liberamente scelti dall'utente. La situazione vista nell'esempio, pur essendo piuttosto ricorrente, non esaurisce la totalità dei casi possibili. Infatti può presentarsi la necessità di operare su costanti piuttosto che su variabili; oppure può essere necessario prelevare le informazioni da elaborare da strutture di dati, per esempio da una tabella.

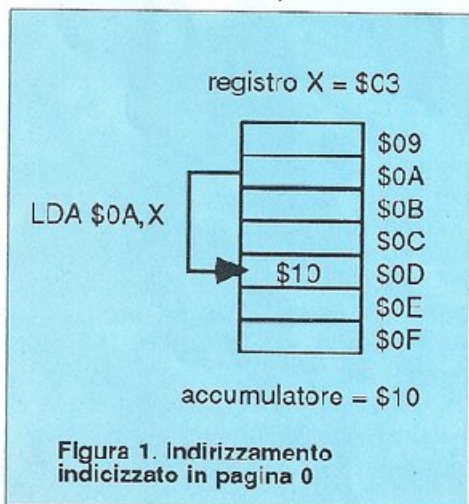
In entrambi i casi bisogna poter modificare la natura dell'istruzione in modo tale che questa tratti il dato (l'operando) nella maniera corretta. La soluzione a questo problema, del tutto generale come vedremo trattando il set di istruzioni del 6510, consiste nel

modo di indirizzamento. Il modo di indirizzamento è un attributo che specifica come l'istruzione dovrà trattare il proprio operando e influisce sul codice dell'istruzione stessa. Per esempio, nel caso della scorsa puntata si potrebbe fare in modo che nell'accumulatore venga caricata la costante 40 e a questa sommata un'altra costante, per esempio 15. A questo scopo l'istruzione LDA 828 andrà modificata in LDA #40 e ADC 829 in ADC #15. Il carattere # sta a indicare una costante: ne parleremo descrivendo l'assemblatore.

I codici operativi delle due istruzioni diventano, rispettivamente, A9 e 69 (in esadecimale). L'indirizzamento utilizzato per le due istruzioni appena viste viene chiamato immediato, proprio perché l'operando costituisce già il dato da elaborare. L'istruzione LDA si presta molto bene come esempio per mostrare gran parte degli indirizzamenti disponibili.

i due byte che seguono l'istruzione devono essere interpretati come un indirizzo da cui prelevare o memorizzare i dati.

2) **indirizzamento in pagina zero:** LDA \$FF Carica in accumulatore il contenuto della locazione FF. Si assembla in A5 FF. Questo tipo di indirizzamento è fondamentalmente identico a quello precedente. In questo caso, però, solo il primo byte seguente l'istruzione viene interpretato come indirizzo. Utilizzando l'indirizzamento in pagina zero si guadagna un byte rispetto all'indirizzamento assoluto ma si restringe il campo d'azione dell'istruzione alla pagina zero (infatti con un byte si possono rappresentare solo i numeri compresi fra 0 e 255). L'indirizzamento in pagina zero può anche essere sostituito da quello assoluto. Per esempio LDA \$FF si può



Gli indirizzamenti

Vediamo quindi quali sono gli altri indirizzamenti utilizzabili con questa istruzione.

1) **indirizzamento assoluto:** LDA \$0400 (il carattere \$ indica un numero espresso in notazione esadecimale) Carica in accumulatore il contenuto della locazione 0400 (1024 in decimale). Si assembla in AD 00 40. Questo modo di indirizzamento è stato usato nell'esempio della puntata precedente dalle istruzioni LDA, ADC e STA. Indica al microprocessore che



modificare in LDA \$00FF e quindi assemblare in AD FF 00. L'istruzione eseguirà la stessa operazione. Tuttavia, viene sprecato un byte e l'esecuzione risulterà più lenta. Questo tipo di indirizzamento non può essere sfruttato largamente sul Commodore 64 in condizioni normali, poiché gran parte della pagina zero viene sfruttata dal sistema operativo.

3) **indirizzamento indicizzato in pagina zero:** LDA \$0A,X e si assembla in B5 0A. Questo tipo di indirizzamento costituisce la soluzione al recupero di dati strutturati in tabelle. All'operando, di un byte, viene aggiunto il contenuto del registro indice X e il risultato costituisce l'indirizzo vero e proprio da cui verrà prelevato il dato. Da notare che non tutte le istruzioni utilizzano lo stesso registro indice come contatore. Fate quindi riferimento alle tre tabelle delle istruzioni. Il concetto di indirizzamento indicizzato in pagina zero viene illustrato in figura 1.

4) **indirizzamento assoluto indicizzato:** LDA \$0400,X (oppure LDA \$0400,Y) e si assembla in BD 00 04 (oppure in B9 00 04). Questo modo di indirizzamento costituisce la soluzione generale al problema di cui si è parlato in precedenza, cioè quello del reperimento di dati strutturati in tabelle. Concettualmente è del tutto identico all'indirizzamento precedente. Rispetto a esso l'indirizzamento assoluto indicizzato risulta più flessibile. Infatti si hanno a disposizione due byte per specificare l'indirizzo base (quindi si può accedere a tutta la memoria disponibile). Inoltre è possibile utilizzare indifferentemente uno dei due registri indice. La figura 2 mostra anche questo tipo di indirizzamento.

5) **indirizzamento indiretto indicizzato** (o indiretto preindicizzato):

LDA (\$0B,X) e si assembla in A1 0B. Questo tipo di indirizzamento consente di gestire dei vettori di puntatori. In particolare può servire per il passaggio di parametri fra sottoprogrammi. Vediamo come funziona. Al valore specificato (in questo esempio 0B) viene aggiunto il contenuto del registro indice X. Il risultato viene interpretato come un indirizzo, a partire dal quale vengono prelevati due byte che formano l'indirizzo effettivo su cui deve agire l'istruzione. Questo modo di indirizzamento ha due caratteristiche importanti: può essere usato solo con il registro indice X e l'indirizzo ottenuto sommando il registro indice X all'indirizzo specificato deve essere in pagina zero. In figura 3 si illu-

stra il concetto di indirizzamento indiretto indicizzato insieme alle locazioni in pagina zero utilizzabili.

6) **indirizzamento indicizzato indiretto** (o indiretto post-indicizzato): LDA (\$0A),Y e si assembla in B1 0A. L'indirizzamento indicizzato indiretto è del tutto simile a quello indiretto indicizzato con la sola differenza che in questo caso il contenuto del registro indice Y viene sommato all'indirizzo contenuto nelle due locazioni contigue di pagina zero e non all'indirizzo specificato nell'istruzione. Anche in questo caso il registro indice utilizzabile è uno solo: quello Y. In figura 4 viene mostrato questo tipo di indirizzamento. Insieme all'istruzione LDA sono disponibili altre istruzioni che permettono il trasferimento di dati, sia tra registri che tra registri e memoria. Vediamole:

TAX copia il contenuto dell'accumulatore nel registro indice X
 TAY copia il contenuto dell'accumulatore nel registro indice Y
 TYA copia il contenuto del registro indice Y nell'accumulatore
 TXA copia il contenuto del registro indice X nell'accumulatore
 TXS copia il contenuto del registro indice X nel puntatore allo stack
 TSX copia il contenuto del puntatore allo stack nel registro indice X

Per queste istruzioni è disponibile un solo tipo di indirizzamento, assente per l'istruzione LDA, l'indirizzamento implicito. Il nome dell'indirizzamento deriva dal fatto che queste istruzioni non hanno bisogno di un operando poiché agiscono esclusivamente sui registri interni del microprocessore. In figura 5 vengono riportati i codici di queste istruzioni. Da notare che le istruzioni di trasferimento tra registri non modificano il contenuto del registro sorgente ma si limitano a copiarne il contenuto in un altro registro. Per esempio, se nel registro indice X si trova il valore 15, eseguendo l'istruzione TXA tale valore verrà copiato nell'accumulatore ma il registro X conterrà ancora 15. Abbiamo visto come sia possibile caricare il principale registro interno, cioè l'accumulatore, con una costante o con il contenuto di una locazione di memoria. Inoltre si è visto anche che è possibile trasferire il contenuto di questo registro in una locazione di memoria, utilizzando l'istruzione STA. Il 6510 consente di eseguire operazioni analoghe anche con altri registri interni. Vediamo quali sono queste istruzioni.

LDX carica il registro indice X
 LDY carica il registro indice Y
 STX copia il registro indice X
 STY copia il registro indice Y

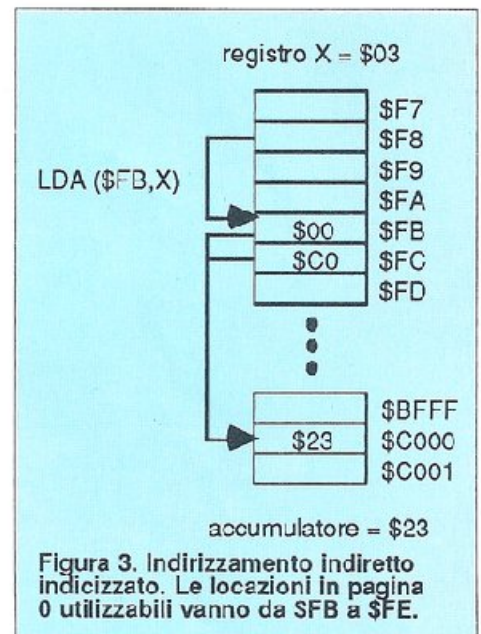


Figura 3. Indirizzamento indiretto indicizzato. Le locazioni in pagina 0 utilizzabili vanno da \$FB a \$FE.

In figura 6 vengono riportati i tipi di indirizzamento disponibili per queste istruzioni e i relativi codici operativi. Si può notare che per le istruzioni di trasferimento da registro a memoria (STA STY STX) non è disponibile l'indirizzamento immediato, cioè l'operando di queste istruzioni non può essere una costante numerica. Il motivo dovrebbe essere abbastanza chiaro. Ogni istruzione di questo tipo fa riferimento al contenuto di un registro interno e quindi sarebbe assurdo tentare di memorizzarne il valore in una costante numerica.

Parlando dell'indirizzamento indiretto e di quello indicizzato si è visto che in entrambi i casi è necessario utilizzare un registro indice. Il contenuto di questo registro viene sommato a un indirizzo per fornire l'indirizzo effetti-

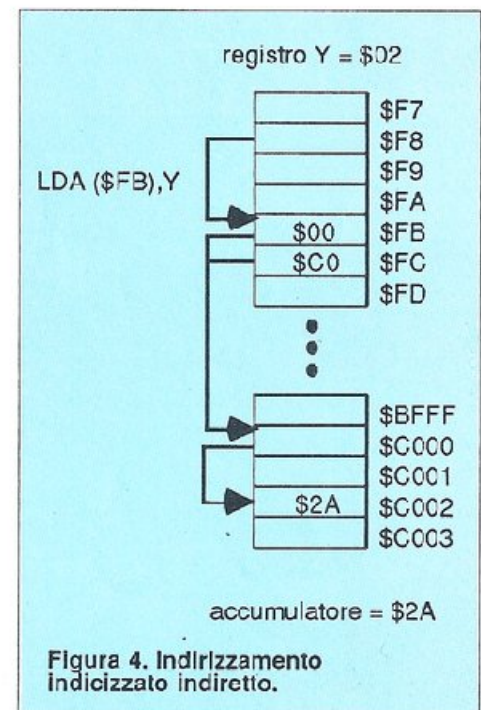


Figura 4. Indirizzamento indiretto indicizzato.

Modo di indirizzamento	Codice mnemonico	Codice operativo	Flag modificati
implicito	TAX	\$AA	NZ ----
implicito	TAY	\$A8	NZ ----
implicito	TYA	\$98	NZ ----
implicito	TXA	\$8A	NZ --
implicito	TXS	\$9A	-----
implicito	TSX	\$BA	NZ ----

Figura 5. Istruzioni di trasferimento fra i registri interni.

vo su cui l'istruzione deve agire. Dovrebbe essere spontaneo chiedersi se sia possibile modificare il contenuto dei registri indice in maniera più efficiente di quanto non si possa fare con le istruzioni di caricamento (LDY LDX). In tal caso si disporrebbe di un mezzo molto potente per agire su dati strutturati in tabelle. Effettivamente questa possibilità esiste. Le istruzioni sono:

INX incrementa di 1 il registro indice X

INY incrementa di 1 il registro indice Y

DEX decrementa di 1 il registro indice X

DEY decrementa di 1 il registro indice Y

Oltre a queste istruzioni specifiche per i registri interni ne esistono altre per manipolare in modo analogo il contenuto delle locazioni di memoria. Vediamole:

INC incrementa di 1 il contenuto di una locazione di memoria

DEC decrementa di 1 il contenuto di una locazione di memoria

In figura 7 vengono riportate tutte le informazioni relative a queste istruzioni. Le istruzioni di incremento

e decremento hanno un comportamento particolare. Se il valore da incrementare è 255, una istruzione di incremento porterà tale valore a 0. Analogamente, se si cerca di decrementare 0 si ottiene 255. Un altro comportamento apparentemente insolito delle istruzioni di incremento e decremento viene mostrato in figura 8. Nell'esempio il registro indice X viene inizialmente caricato con il valore 0. Quindi vengono eseguite più istruzioni di decremento. Si può notare che il flag N è posto a 1 finché il contenuto del registro indice non diventa inferiore a 128. Fino a quel momento il contenuto del registro indice viene segnalato negativo. Tuttavia se usato in un indirizzamento indicizzato o indiretto il valore del registro indice verrebbe considerato positivo. Per esempio:

LDX #FF N=1

LDA COO,X

La prima istruzione carica nel registro indice X il valore decimale 255. Il flag del segno, proveniente dal bit 7 dell'ultimo valore trattato, è settato: il numero è considerato negativo poiché il bit del segno è a 1 (infatti la rappresentazione binaria di 255 è 11111111). Tuttavia l'istruzione suc-

cessiva caricherà in accumulatore il contenuto della locazione COFF. Da quanto si è visto risulta indispensabile disporre di una serie di istruzioni che consentano di verificare lo stato dei flag del registro di stato e permettano di agire di conseguenza (fate riferimento alle tabelle riportate nelle figure per sapere quali flag vengono modificati da ciascuna istruzione). Per ora ci limiteremo a introdurre due sole istruzioni di questo tipo:

- BEQ (da Branch on Equal=salta se uguale) effettua un test sul flag Z. Se questo è settato, cioè è posto a 1, allora opera un salto all'indirizzo specificato. In caso contrario l'esecuzione del programma prosegue con l'istruzione immediatamente successiva.

- BNE (da Branch if Not Equal=salta se non è uguale) effettua un test sul flag Z. Se Z=0 allora opera un salto all'indirizzo specificato. Anche in questo caso se l'esito del test risulta negativo l'esecuzione del programma riprende con l'istruzione successiva. Entrambe le istruzioni (figura 9) dispongono di un solo modo di indirizzamento: l'indirizzamento relativo. Il nome deriva dal fatto che l'indirizzo a cui deve avvenire il salto se il test effettuato ha esito positivo non viene indicato esplicitamente ma viene espresso come spostamento da compiere rispetto alla locazione in cui si trova l'istruzione stessa. Lo spostamento deve essere espresso in complemento a due (infatti si possono avere sia salti negativi, cioè all'indietro, sia salti positivi, cioè in avanti), utilizzando un solo byte, e viene calcolato a partire dall'indirizzo della prima istruzione successiva a quella che effettua il test. In figura 10 e 11 viene mostrato un esempio.

L'assemblatore

Anche avendo a disposizione le tabelle con i codici delle istruzioni, compiere l'assemblaggio a mano risulta molto noioso e soprattutto estremamente pericoloso: un solo errore nel tradurre le istruzioni può compromettere l'intero lavoro. Per evitare questi inconvenienti e soprattutto per consentire una programmazione lineare e leggibile si ricorre all'assembler (da cui il nome del linguaggio di programmazione: assembler simbolico). L'assemblatore, infatti, permette di scrivere programmi in assembler simbolico e quindi si occupa di tradurli in codice macchina vero e proprio. Poiché d'ora in poi useremo sempre questo strumento per scrivere i programmi d'esempio cominceremo a descriverne il funzionamento. L'assembler cui fare-

Indirizzamento	LDA	LDX	LDY	STA	STX	STY
Immediato	\$A9	\$A2	\$A0	-	-	-
Pagina zero	\$A5	\$A6	\$A4	\$85	\$86	\$84
Pagina zero,X	\$B5	-	\$B4	\$95	-	\$94
Pagina zero,y	-	\$B6	-	-	\$96	-
Assoluto	\$AD	\$AE	\$AC	\$8D	\$8E	\$8C
Assoluto,X	\$BD	-	\$BC	\$9D	-	-
Assoluto,Y	\$B9	\$BF	-	\$99	-	-
(Indiretto,X)	\$A1	-	-	\$81	-	-
(Indiretto),Y	\$B1	-	-	\$91	-	-

Figura 6. Istruzioni di trasferimento registro-registro, registro-memoria. LDA, LDX, LDY modificano i flag N e Z, mentre le altre istruzioni lasciano i flag invariati.

mo riferimento è il Macro Assembler Development System della Commodore.

Questo assembler si compone di più programmi: un editor, un assembler, due monitor e due caricatori. L'editor è ciò che consente di scrivere un programma utilizzando il formato simbolico visto in precedenza. L'assembler si occupa di tradurre il programma creato con l'aiuto dell'editor, e detto codice sorgente, in codice macchina, e di memorizzare quest'ultimo su disco. Una volta che l'assembler ha completato l'operazione di traduzione, è necessario utilizzare il caricatore (nel pacchetto software di cui stiamo parlando ce ne sono due e occupano due diverse zone di memoria. In questo modo il programmatore ha una maggiore scelta per l'area di memoria in cui porre il proprio programma) per inserire in memoria il codice oggetto. Una volta in memoria il programma può essere eseguito. Il monitor serve per la messa a punto del programma e permette di agire direttamente sul codice oggetto già inserito in memoria. Anche per questo programma si è pensato bene di fornire due versioni che occupano diverse aree di memoria. Per ora ci occuperemo dell'assembler e dell'editor. D'ora in poi quando useremo la parola assembler indicheremo l'insieme costituito dall'editor e dall'assembler.

Rispetto al formato normale delle istruzioni, cioè quello visto la scorsa volta nel programma di somma, l'assembler offre 2 campi in più. Insieme al codice mnemonico e all'operando sono disponibili i campi Label e Commento. Il formato di una istruzione in assembler simbolico sarà così la seguente: (label) mnemonico (operando) (commento). Le parentesi indicano che il campo è opzionale. Analizziamo tutti i campi.

• **Label** (=etichetta). Questo campo, sempre opzionale, consente di dare un nome a una istruzione del programma. Così facendo, ogni volta che nel programma si vuole fare riferimento a questa istruzione basta specificare nel campo operando il nome a essa associato. Le restrizioni poste per la label sono le seguenti:

- deve iniziare con un carattere alfanumerico.
- deve essere composta da un massimo di 6 caratteri.
- non può essere né un codice mnemonico né uno dei seguenti caratteri A, S, P, X o Y.

• **Mnemonico**. Di questo campo si è già parlato ampiamente e quanto è stato detto rimane comunque valido anche utilizzando l'Assembler.

Indirizzamento	INX	INY	INC	DEX	DEY	DEC
Implicito	SE8	\$C8	-	\$CA	\$88	-
Pagina zero	-	-	\$E6	-	-	\$C6
Pagina zero,X	-	-	\$F6	-	-	\$D6
Absolute	-	-	\$EE	-	-	\$CE
Absolute,X	-	-	\$FE	-	-	\$DE

Figura 7. Istruzioni di incremento e decremento. Tutte le istruzioni modificano i flag N e Z.

• **Operando**. L'indirizzo o il valore numerico su cui deve agire l'istruzione possono essere espressi in vari modi. Per specificare un dato si deve premettere al numero il carattere #. Quindi il numero può essere espresso:

- *In esadecimale*. In questo caso il numero deve essere preceduto dal simbolo \$. Per esempio LDA #\$A0 carica in accumulatore il valore A0 (160 decimale)

- *In binario*. Allora il numero deve essere preceduto dal simbolo %. Per esempio LDX #%00000001 carica nel registro indice X il valore 00000001 (1 decimale)

- *In decimale*. In questo caso il numero non deve essere preceduto da nessun simbolo. Per esempio LDY #45 carica nel registro indice il valore 45.

Nel caso che il dato da elaborare sia un codice ascii, il carattere, di cui si vuole il codice, si fa precedere dal simbolo " " (apice singolo, codice ascii 39). Per esempio LDA #'E carica in accumulatore il codice ascii del carattere E (69).

Istruzione	registro X	flag N
LDX #0	0	0
DEX	255	1
DEX	254	1
DEX	253	1
...
DEX	127	0
DEX	126	0
...
DEX	0	0

Figura 8. Effetti dell'istruzione DEX sul flag N.

Indirizzamento	BEQ	BNE
Relativo	\$F0	\$D0

Figura 9. Istruzioni di salto condizionato. Entrambe le istruzioni lasciano tutti i flag invariati.

Una costante numerica può anche essere indicata con un simbolo (in seguito vedremo in che modo e quando devono essere definiti i simboli). Per esempio LDA #MESE carica un accumulatore il valore associato al simbolo MESE.

Nel caso che l'operando sia un indirizzo valgono tutte le convenzioni stabilite per le costanti, a eccezione del carattere # che deve essere omesso. Inoltre così come si era fatto nel pro-

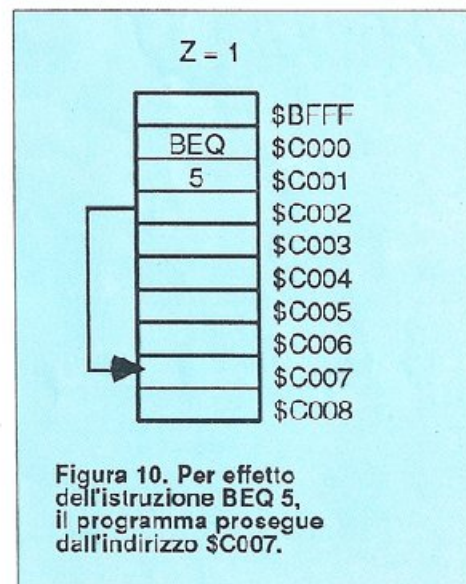


Figura 10. Per effetto dell'istruzione BEQ 5, il programma prosegue dall'indirizzo \$C007.

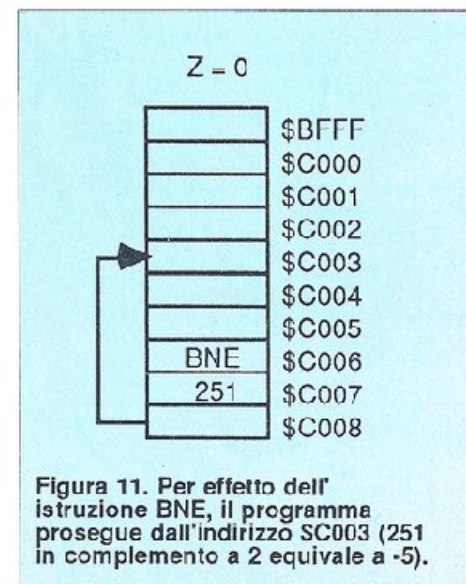


Figura 11. Per effetto dell'istruzione BNE, il programma prosegue dall'indirizzo \$C003 (251 in complemento a 2 equivale a -5).

gramma di somma, anche utilizzando l'assembler gli indirizzi dovranno essere scritti nel formato byte alto-byte basso, si occuperà l'assemblatore di tradurli nel formato corretto. LDA \$A000, per esempio, carica in accumulatore il contenuto della locazione di memoria A000 (40960 decimale), LDA SCHRM carica in accumulatore il contenuto dell'indirizzo associato al simbolo SCHRM.

• **Commento.** Questo campo, sempre opzionale, consente di docu-

mentare un programma. Per poter inserire un commento basta digitare il carattere ";" e quindi inserire il messaggio.

Il carattere ";" deve essere separato dal campo operando (o da quello mnemonico) da almeno uno spazio.

Le direttive

Una importante facilitazione alla programmazione offerta dall'assembler è rappresentata dalle direttive. Lo

scopo principale delle direttive consiste nel riservare aree di memoria per i dati. Analizziamo ora le direttive più importanti fra quelle disponibili sul macro assembler della Commodore.

• = Questa direttiva serve per associare un valore numerico a un simbolo (per i simboli valgono le stesse osservazioni fatte per la label). Per esempio con BASE=\$A000 si associa al simbolo BASE il valore A000, l'assegnamento di un simbolo può anche essere fatto sulla base di una

Figura 12.

```

00001 0000 ;QUESTO PROGRAMMA COSTITUISCE UN
00002 0000 ;ESEMPIO DI UTILIZZO DELL'INDIRIZ-
00003 0000 ;ZAMENTO INDICIZZATO. TRASFERISCE
00004 0000 ;LE PRIME 240 LOCAZIONI DELLA MEMO-
00005 0000 ;RIA VIDEO
00006 0000 ;
00007 0000 *=$C000 ;ASSEMBLA A PARTIRE DA $C000
00008 0000 VID1=1024 ;PRIMA LOCAZIONE VIDEO
00009 0000 VID2=1424 ;LOCAZIONE VIDEO A PAR-
00010 0000 ;TIRE DALLA QUALE VERRA'
00011 0000 ;MEMORIZZATO IL TESTO.
00012 0000 ;LOCAZIONE DELLA MEMORIA
00013 0000 ;COLORE CORRISPONDENTE ALLA
00014 0000 ;LOCAZIONE 1424
00015 0000 A0 01 LDY #1 ;PONE NEL REGISTRO Y IL CODICE
00016 0002 ;DEL COLORE (1=BIANCO)
00017 0002 A2 F0 LDY #240 ;INIZIALIZZA IL REGISTRO X
00018 0004 ;PER ESSERE USATO COME CONTRA-
00019 0004 ;TORE
00020 0004 98 CICLO TYA ;COPIA IL CONTENUTO DEL
00021 0005 ;REGISTRO Y NELL'ACCUMULATORE
00022 0005 30 90 03 STA COLORE,X ;COPIA IL CONTENUTO
00023 0008 ;DELL'ACCUMULATORE NELLA
00024 0008 ;MEMORIA DEL COLORE
00025 0008 BD 00 04 LDA VID1,X ;CARICA IN ACCUMULATORE
00026 0008 ;IL CONTENUTO DELLA LOCAZIO-
00027 0008 ;NE DA TRASFERIRE
00028 0008 30 90 05 STA VID2,X ;EFFETTUA IL TRASFERIMEN-
00029 000E ;TO
00030 000E DA DEY ;DECREMENTA IL REGISTRO X PER
00031 000F ;PASSARE A UN ALTRA LOCAZIONE
00032 000F D0 F3 BNE CICLO ;CONTROLLA SE SONO GIA'
00033 0011 ;STATI TRASFERITI I CONTENUTI
00034 0011 ;DELLE 239 LOCAZIONI DEL VIDEO
00035 0011 ;SEGUENTI LA LOCAZIONE 1024
00036 0011 ;SE SI SALTA A CICLO E CONTI-
00037 0011 ;NUA IL TRASFERIMENTO
00038 0011 ;IN CASO CONTRARIO
00039 0011 96 TYA ;COPIA NELL'ACCUMULATORE IL RE-
00040 0012 ;GISTRO Y
00041 0012 30 90 03 STA COLORE,X ;TRASFERISCI IL CONTE-
00042 0015 ;NUTO DELL'ACCUMULATORE NELLA
00043 0015 ;MEMORIA DEL COLORE
00044 0015 BD 00 04 LDA VID1,X ;CARICA IN ACCUMULATORE
00045 0018 ;IL CONTENUTO DELLA LOCAZIONE
00046 0018 ;1024 (X CONTIENE 0)
00047 0018 90 90 05 STA VIDEO,X ;E MEMORIZZALO NELLA LO-
00048 001B ;CAZIONE 1424
00049 001B 60 RTS ;RITORNA AL BASIC
00050 001C .END

```


operazione aritmetica: per esempio $BASE = \$A000 + 1$. E' buona regola usare il più spesso possibile simboli piuttosto che valori numerici. Agendo in questo modo si possono creare programmi facilmente modificabili. Infatti basta cambiare il valore assegnato al simbolo perché automaticamente l'assemblatore modifichi gli operandi in cui questo compare. Senza utilizzare i simboli bisognerebbe correggere uno per uno tutti gli indirizzi (o le costanti) che si vogliono modificare. A

differenza di quanto succede in basic, i simboli possono essere utilizzati solo se sono stati precedentemente definiti.

• *= Questa direttiva serve per indicare al programma assemblatore da quale locazione si vuole iniziare ad assemblare. Per esempio *=C000 indica all'assembler che si vuole iniziare ad assemblare il programma a partire dalla locazione C000 (49152 decimale). Questa direttiva deve sempre essere usata se non si vuole che il programma

venga assemblato a partire dalla locazione 0.

• ** Con questa direttiva si possono riservare dei byte di memoria. Ecco un esempio di utilizzo:

*= \$C000

**= *+2

In questo modo si sono riservati due byte a partire dalla locazione C000. Quindi il programma da assemblare partirà dalla locazione C002. Non c'è alcun limite al numero di byte che si possono riservare.

Figura 13.

```

00001 0000 ;QUESTO PROGRAMMA COSTITUISCE UN
00002 0000 ;ESEMPIO DI UTILIZZO DELL'INDIRIZ-
00003 0000 ;ZAMENTO INDIRETTO. TRASFERISCE LE
00004 0000 ;PRIME 800 LOCAZIONI DI MEMORIA
00005 0000 *= $C400 ;ASSEMBLA DA $C400
00006 C400 A9 00 LDA #0 ;INIZIALIZZA I PUNTATORI
00007 C402 85 FB STA $FB ;IN PAGINA ZERO
00008 C404 A9 04 LDA #$4
00009 C406 85 FC STA $FC
00010 C408 A9 00 LDA #0
00011 C40A 85 FD STA $FD
00012 C40C A9 C0 LDA #$C0
00013 C40E 85 FE STA $FE
00014 C410 PUNT1=$FB ;DEFINISCE I SIMBOLI
00015 C410 PUNT2=$FD ;CHE IDENTIFICANO LE LO-
00016 C410 ;CAZIONI IN PAGINA 0 UTI-
00017 C410 ;LIZZATE DALL'INDIRIZZA-
00018 C410 ;MENTO INDIRETTO
00019 C410 A2 03 LDX #3 ;PONI NEL REGISTRO X IL NU-
00020 C412 ;MERO DI BLOCCHI DI 256 BYTE
00021 C412 ;DA TRASFERIRE
00022 C412 A0 00 CL1 LDY #0 ;SETTA IL REGISTRO Y PER
00023 C414 ;ESSERE UTILIZZATO COME CONTA-
00024 C414 ;TORE
00025 C414 B1 FB CL2 LDA (PUNT1),Y ;CARICA IN ACCU-
00026 C416 ;MULATORE IL CONTENUTO DELLA
00027 C416 ;LOCAZIONE PUNTATA DA PUNT1
00028 C416 91 FD STA (PUNT2),Y ;E LO MEMORIZZA NELLA
00029 C418 ;LOCAZIONE PUNTATA DA PUNT2
00030 C418 A9 20 LDA #32 ;CARICA IN ACCUMULATORE
00031 C41A ;IL CODICE VIDEO DELLO SPAZIO
00032 C41A 91 FB STA (PUNT1),Y ;E LO MEMORIZZA NELLA
00033 C41C ;LOCAZIONE PUNTATA DA PUNT1
00034 C41C 88 DEY ;DECREMENTA IL CONTENUTO DI Y
00035 C41D D0 F5 BNE CL2 ;E CONTROLLA SE IL BLOCCO
00036 C41F ;E' GIA' STATO TRASFERITO
00037 C41F E6 FC INC PUNT1+1 ;SE SI, PASSA AL BLOCCO
00038 C421 E6 FE INC PUNT2+1 ;SUCCESSIVO
00039 C423 CA DEX ;DECREMENTA X E CONTROLLA SE
00040 C424 ;SONO STATI TRASFERITI
00041 C424 D0 EC BNE CL1 ;TUTTI I BLOCCHI
00042 C426 A0 20 LDY #32 ;SE SI, TRASFERISCE GLI UL-
00043 C428 ;TIMI 32 BYTE
00044 C428 B1 FB CL3 LDA (PUNT1),Y ;QUESTA PARTE EF-
00045 C42A 91 FD STA (PUNT2),Y ;FFETTUA IL TRASFERI-
00046 C42C A9 20 LDA #32 ;MENTO IN MODO ANALOGO A
00047 C42E 91 FB STA (PUNT1),Y ;PRIMA
00048 C430 88 DEY

```

(Continua)


```

00049 C431 D0 F5      BNE     CL3          ;SE E' TERMINATO IL TRASFE-
00050 C433 60        RTS          ;RIMENTO TORNA AL BASIC
00051 C434          ;QUESTA PARTE RIFORTA SUL VIDEO
00052 C434          ;IL TESTO TRASFERITO
00053 C434 A9 04     LDA     #4          ;RIPRISTINA I PUNTATORI
00054 C436 85 FC     STA     PUNT1+1    ;IN PAGINA ZERO
00055 C438 A9 C0     LDA     #C0
00056 C43A 85 FE     STA     PUNT2+1
00057 C43C A2 03     LDM     #3          ;CARICA IN ACCUMULATORE IL
00058 C43E          ;NUMERO DI BLOCCHI DA TRASFERIRE
00059 C43E A0 00     CL4     LDY #0      ;QUINDI EFFETTUA IL
00060 C440          ;TRASFERIMENTO IN MO-
00061 C440          ;DO ANALOGO A PRIMA UTILIZ-
00062 C440 B1 FD     CL5     LDA (PUNT2),Y ;ZANDO PUNTE
00063 C442 91 FB     STA     (PUNT1),Y  ;COME PUNTATORE AL
00064 C444 80        DEY          ;TESTO DA TRASFERIRE E PUNT1
00065 C445 D0 F9     BNE     CL5          ;COME PUNTATORE ALLA ZONA
00066 C447 E6 FC     INC     PUNT1+1    ;D'ARRIVO
00067 C449 E6 FE     INC     PUNT2+1
00068 C44B CA        DEY
00069 C44C D0 F0     BNE     CL4
00070 C44E A0 20     LDY     #3E
00071 C450 R1 FD     CL6     LDA (PUNT2),Y
00072 C452 91 FB     STA     (PUNT1),Y
00073 C454 88        DEY
00074 C455 D0 F9     BNE     CL6
00075 C457 60        RTS          ;SE IL TESTO E' STATO TRA-
00076 C458          ;FERITO TORNA AL BASIC
.END

```

• **.Byte** Questa è usata per riservare un byte di memoria e riempirlo con un valore specificato. Si possono inserire anche più byte nella stessa direttiva, e in questo caso verranno memorizzati consecutivamente a partire dal-

l'indirizzo specificato, oppure stringhe di caratteri, che dovranno essere racchiuse fra due caratteri "". Vediamo qualche esempio:

*=\$C000
.BYTE \$FF,%00001010,5

.BYTE 'ELLE'

La prima direttiva segnala all'assemblatore che il programma inizia dalla locazione C000 esadecimale. Le due direttive seguenti memorizzano a partire da quella locazione di memoria i va-

CRUCISPECTRUM

Soluzione dello schema pubblicato sul numero scorso

G	R	A	N	P	R	E	M	I	O	D	I	S	A	N	M	A	R	I	N	O
R	A	D	I	O	E	S	T	E	T	I	S	T	I	A	E	R	V			
A	Z	U	L	E	N	E	S	R	A	U	D	M	A	R	V					
N	I	L	O	T	I	C	O	E	P	I	S	T	E	M	A	T	I	C	I	
S	O	A	E	R	S	I	A	M	A	T	O	L	O	G	I	A				
A	N	T	E	S	I	G	N	A	N	O	U	N	T	I	A	C	N			
S	A	O	S	U	R	A	L	I	M	E	N	T	A	R	E	T	U	T		
S	T	R	I	A	T	U	R	A	B	T	E	T	E	A	T	E	T	E		
O	A	I	O	A	E	M	B	O	L	O	O	U	V	A						

lori decimali 255, 10 e 5 e i caratteri che compongono la stringa "ELLE".

• **.WORD** Viene usata per riservare e caricare due locazioni di memoria consecutive con due byte di dati. Non è possibile inserire caratteri alfanumerici in questa direttiva. **.WORD** memorizza i dati a due byte nell'ordine byte basso-byte alto. Per questo può essere utilizzata per la creazione di indirizzi da utilizzare con istruzioni che sfruttano l'indirizzamento indiretto. Vediamo un esempio:

```
*=SC000
.WORD SFFA1
```

La seconda direttiva memorizza a partire dalla locazione C000 i valori A1 e FF.

• **.DBYTE** E' del tutto simile alla direttiva precedente. L'unica differenza consiste nel fatto che i dati vengono memorizzati nell'ordine byte alto-byte basso. Per esempio, sostituendo la direttiva **.WORD** con **.DBYTE** nell'e-

Figura 14.

```
1 REM -----
2 REM INDIRIZZAMENTO INDICIZZATO
3 REM -----
4 :
100 POKE 53200,0:POKE53201,0
110 PRINT"□":MV=1024:MC=55296
115 FORT=49152T049179
117 READA:POKET,A:NEXT
120 FORT=0T0239
130 C=INT(RND(0)*25)+1
140 POKEMV+T,C:POKEMC+T,1
150 NEXT
160 PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
170 PRINTTAB(3)"TRASFERIMENTO IN BASIC - PREMI F1"
180 GETA#:IFA#<"■"THEN180
190 PRINT"□"
200 FORT=0T0239
210 POKEMV+400+T,PEEK(MV+T)
220 POKEMC+400+T,PEEK(MC+T)
230 NEXT
240 PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
250 PRINTTAB(13)"PREMI UN TASTO"
260 POKE198,0:WAIT198,1:POKE198,0
270 PRINT"XXXXXXXXXXXXXXXXXXXX"
280 FORI=0T05
290 PRINT"
:POKEMV+439+40*I,32
300 NEXT
310 PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
320 PRINTTAB(3)"TRASFERIMENTO IN L.M. - PREMI F1"
330 GETA#:IFA#<"■"THEN330
340 PRINT"□"
350 SYS49152
360 TI#="000000"
370 IFTI<300THEN370
380 END
1000 DATA 160,1,162,240,152,157,144,217
1010 DATA 189,0,4,157,144,5,202,208,243
1020 DATA 152,157,144,217,189,0,4,157
1030 DATA 144,5,96
```

Figura 15.

```
1 REM -----
2 REM INDIRIZZAMENTO INDIRETTO
3 REM -----
4 :
100 POKE 53200,0:POKE53201,0
110 PRINT"□":MV=1024:MC=55296
115 FORT=50176T050263
117 READA:POKET,A:NEXT
120 FORT=0T0799
130 C=INT(RND(0)*25)+1
140 POKEMV+T,C:POKEMC+T,1
150 NEXT
160 PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
170 PRINTTAB(5)"PER IL TRASFERIMENTO PREMI F1"
180 GETA#:IFA#<"■"THEN180
190 PRINT"□"
200 SYS50176
240 PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
250 PRINTTAB(3)"PER RISCRIVERE IL TESTO PREMI F7"
260 GETA#:IFA#<"■"THEN260
270 PRINT"□"
350 SYS50228
360 TI#="000000"
370 IFTI<300THEN370
380 END
1000 DATA 169,0,133,251,160,4,133,252
1010 DATA 169,0,133,253,169,192,133
1020 DATA 254,162,3,160,0,177,251,145
1030 DATA 253,169,32,145,251,136,208
1040 DATA 245,230,252,230,254,202,208
1050 DATA 236,160,32,177,251,145,253
1060 DATA 169,32,145,251,136,208,245
1070 DATA 96,169,4,133,252,169,192,133
1080 DATA 254,162,3,160,0,177,253,145
1090 DATA 251,136,208,249,230,252,230
1100 DATA 254,202,208,240,160,32,177
1110 DATA 253,145,251,136,208,249,96
```

sempio precedente a partire dalla locazione C000 verrebbero memorizzati i valori FF e A1.

• **.END** Questa direttiva segnala all'assembler che il programma è terminato. Deve sempre essere posta alla fine di ogni programma.

Due dimostrativi

I due programmi che seguono costituiscono una applicazione pratica di gran parte di quello che si è visto fino a ora.

In particolare mostrano l'estrema potenza degli indirizzamenti indicizzato e indiretto.

In figura 12 e 13 vengono riportati i listati, prodotti dall'assembler, dei due programmi. Sono ampiamente documentati e dovrebbe risultare abbastanza chiaro il loro funzionamento. In figura 14 e 15 riportiamo i relativi caricatori basic con cui è possibile mandarli in esecuzione.

Paolo Gussoni

Vorrei sapere, vorrei proporre...

A proposito di animazione

Sono un vostro lettore, abbonato da poco e molto interessato al vostro mensile. Scrivo per chiedervi di mandarmi alcune indicazioni riguardanti la costruzione di figure animate col C64. Non sono ancora riuscito a capire come si fa a entrare in alta risoluzione o come si possa disegnare sul video: aerei, automobili di diversa grandezza e altri tipi di figure. Vi sarei molto riconoscente se mi forniste delle indicazioni di programmi e routine che spieghino come si può accedere alle locazioni di memoria che sovrintendono alla grafica.

Gianfranco De Gasperi
Valdidentro (SO)

Lo spazio solitamente dedicato alle lettere sarebbe decisamente insufficiente per esaurire dettagliatamente un argomento così vasto come la grafica, su cui d'altra parte sono stati scritti interi volumi. In queste poche righe ci limiteremo a dare delle indicazioni di fondo, rimandando a testi e programmi per il necessario approfondimento, nonché al corso a puntate che pubblicheremo tra pochi mesi. Per disegnare sullo schermo esistono diversi modi, la scelta dei quali dipende essenzialmente dalle caratteristiche dell'immagine, dall'uso che si vuole farne e dal tempo che si è disposti a sacrificare per produrla:

1. **Bassa risoluzione e caratteri grafici.** In questo modo si può fare ben poco, anche se il C64 mette a disposizione un buon numero di tasti grafici le figure con un minimo di complessità non possono venir riprodotte con sufficiente fedeltà.

2. **Bassa risoluzione e caratteri in extended background.** Consente di visualizzare caratteri con uno sfondo locale diverso da quello della pagina. Purtroppo in questo modo non è possibile visualizzare i caratteri grafici e quindi questo modo è utile solo per evidenziare testi e messaggi.

3. **Bassa risoluzione e caratteri ridefiniti.** Una tecnica già più efficace è quella di rimodellare i caratteri grafici per le esigenze specifiche del programma cui devono servire. Per fare questo è necessario copiare dalla ROM i codici che descrivono le immagini dei caratteri che si vogliono utilizzare (sia quelli da modificare sia quelli da mantenere invariati, come i testi) e di andare a ridefinirli sostituendo a essi la nuova immagine. Quindi basta informare il C64 che deve leggere tali immagini dalla RAM e non più dalla ROM e il gioco è fatto. Detto così sembra facile, in realtà occorre un lavoro da certissimo per ridisegnare tutti i singoli caratteri e spesso per assemblare una figura di grosse dimensioni ne

occorrono molti. E' indispensabile un buon programma "disegnatore" come quello pubblicato sul numero scorso di REC (aprile 86, Sprite & Character editor) per ridurre i problemi della conversione delle immagini in forma binaria e viceversa. Tuttavia anche con tutti gli strumenti e la buona volontà questa tecnica non può arrivare ovunque. Figure a pieno schermo o animazioni certamente non potrebbero essere prodotte. Inoltre il colore potrebbe essere cambiato indipendentemente solo su rettangoli di 8x8 pixel.

4. **Sprite monocromatici, multicolore, normali ed espansi.** Gli sprite sono oggetti di dimensioni 24x21 pixel nel modo normale e di 48x42 nel modo espanso, possono essere spostati liberamente sullo schermo senza interferire con la bassa o l'alta risoluzione presente sullo sfondo e possono avere due diverse soluzioni cromatiche: 1 colore e 3 colori (oltre a quello dello sfondo). Questi oggetti vengono definiti per mezzo di un'immagine binaria posta in una zona di memoria RAM del tutto simile (eccetto che per le dimensioni) a quella dei caratteri. Gli sprite sono indicati per descrivere figure di media grandezza e in movimento. Vengono governati per mezzo di puntatori posti subito dopo la memoria video e possono essere riposti quasi ovunque nella RAM. Tuttavia sullo schermo ne possono essere visualizzati al massimo 8 contemporaneamente, per questo vanno utilizzati con molta parsimonia e laddove è indispensabile il movimento. Anche in questo caso è fondamentale avere un programma "disegnatore" come Sprite & Character editor.

5. **Alta risoluzione monocromatica e multicolore.** L'alta risoluzione diventa indispensabile quando è necessario indirizzare il singolo punto sullo schermo e la figura è troppo grande per essere composta da caratteri o sprite oppure quando la figura non è predefinita, ma per esempio è funzione dell'elaborazione: grafici tridimensionali o a torta, studi di funzioni a una o a due dimensioni, schermate di presentazione, eccetera. Bisogna anche qui distinguere tra modo monocromatico e multicolore: il primo consente una risoluzione di 320x200 punti e 1 solo colore di linea, il secondo consente una risoluzione di 160x200 punti e 3 colori di linea. L'alta risoluzione porta via molta memoria ed è consigliabile utilizzarla solo quando è veramente indispensabile; infatti si incontrano diverse difficoltà: il disegno deve essere effettuato da un programma scritto in linguaggio macchina (a meno di non voler attendere ore per vederlo visualizzato completamente) e solo quando è completo può essere utilizzato da Basic. Ci sono molti tool in circolazione adatti a questo scopo e possono es-

sere supportati da diversi hardware come tavolette grafiche, penne ottiche, paddle e joystick: Koala Painter, Blazing paddles, Flexi Draw, Paint Magic, Print Shop, e altri ancora. La più grossa limitazione dell'alta risoluzione consiste nell'impossibilità di utilizzarla contemporaneamente al testo, almeno senza usare espansioni grafiche come il Simon's Basic, l'Ultrabasic o il Video Basic e senza ricorrere a routine da interruzione in linguaggio macchina.

Come vede, il panorama delle scelte possibili è molto ampio, e ogni soluzione ha le sue controindicazioni. La strada più facile è comunque quella delle espansioni grafiche, specialmente se si dispone di un disk drive: tra queste segnaliamo senza dubbio Basic Lighing della Oasis software.

Per maggiori informazioni riguardanti la grafica e le tecniche di implementazione di routine grafiche le consigliamo: Guida al Commodore 64, Mc Graw Hill, 430 pagine, 35.000 lire e, se conosce l'inglese, Commodore 64, Graphics and Sound, Granada, 196 pagine, 24.000 lire.

Come simulare Print at?

Nel Basic del C64 non è previsto il comando PRINT AT, che consente di visualizzare l'oggetto dell'istruzione PRINT alle coordinate desiderate. Su altri computer questo comando invece è presente (Spectrum). Vorrei chiedervi come è possibile simulare questa funzione senza ricorrere a espansioni o a routine in linguaggio macchina.

Paolo Garlini - Bologna

L'inconveniente della povertà del Basic del C64 in questo caso è facilmente superabile. Ci sono due possibilità: o si definisce una stringa contenente il posizionamento in HOME (codice ASCII 19) più 25 caratteri di movimento verticale (codice ASCII 17) e la si utilizza per il posizionamento verticale (per il posizionamento orizzontale c'è il comando TAB), oppure si modificano le due locazioni che contengono la posizione del cursore sullo schermo e si richiama con una SYS la routine del sistema operativo (SYS 58640). Entrambe sono valide: l'unica differenza è che la prima va usata direttamente nel comando PRINT, la seconda indipendentemente. Ecco due esempi:

```
100 PRINTCHR$(147)
110 Y$=CHR$(19)
120 FO-
RI=I*O25.Y$=Y$+CHR$(17).NEXT
130 PRINTLEFT$(Y$,13)TAB(15)"EDITRONICA"
140 END
```

```
100 PRINTCHR$(147)
110 POKE 214,12
```


120 POKE 211,15
130 SYS 58640
140 PRINT"EDITRONICA"
150 END

Entrambe le routine stampano il messaggio "EDITRONICA" centrato sullo schermo.

Un computer per la sera

Ho la licenza media; qualche anno fa, per motivi di lavoro, ho dovuto abbandonare gli studi. Ora però vorrei riprenderli, possibilmente in una scuola serale, per conseguire un diploma. Nel frattempo, vorrei continuare a interessarmi attivamente di computer per mantenermi al passo con i tempi. Potete fornirmi qualche indicazione in merito?

Paolo Bardi - Muggiò (MI)

Probabilmente entro pochi anni molte scuole medie superiori statali utilizzeranno regolarmente il computer nella didattica, in conformità al piano nazionale per l'introduzione dell'informatica nelle scuole secondarie superiori. Possiamo indicare nel frattempo, tra le scuole statali della sua zona l'I.T.I.S. serale di Sesto S. Giovanni, in via Leopardi 22, tel. 02/2403441, al quale si accede con la licenza media.

La scuola, che forma meriti meccanici, elettrotecnici e informatici, è dotata di ottime attrezzature e di aule appositamente predisposte per l'apprendimento fin dal primo anno di elementi di programmazione con uso di computer da parte degli studenti. Quest'ultima attività si svolge all'interno di singole discipline di studio senza tagli ai normali programmi.

L'orario delle lezioni è il seguente: dal lunedì al venerdì dalle 18.30 alle 22.00; sabato dalle 14 alle 18. Le domande di iscrizione si ricevono dal 23 giugno al 7 luglio di ogni anno.

Un chiarimento?
Un problema? Un'idea?
Scriveteci.

Gli esperti di Radio
Elettronica & Computer
sono a vostra
disposizione per
qualsiasi quesito.
Indirizzate a:
RadioElettronica &
Computer LETTERE
Corso Monforte, 39
20122 Milano.

Potente e capace il nuovo linguaggio

Posseggo un C64 da diversi anni e ormai conosco molto bene sia il Basic sia il linguaggio macchina del 6510. Mi è capitato di provare a fare esperienza con altri linguaggi di programmazione come il tanto decantato Pascal e il Forth, tuttavia sono rimasto abbastanza deluso. In particolare il Pascal, per esigenze di compilazione mi costringe a una debug lunga e laboriosa, aggravata dalla lentezza del drive 1541. Ho letto su alcune riviste americane della pubblicità di un nuovo linguaggio di programmazione: il Promal. Vorrei saperne di più.

Stefano Marapelli - Firenze

Il Promal (PROgrammer's Micro Application Language) è un linguaggio ad alto livello compilato progettato da Bruce D. Cabrey su Copyright della Systems Management Associates. Sintatticamente è una via di mezzo tra il Pascal e il C; possiede quindi tutta la potenza di un linguaggio ad alto livello strutturato e la capacità di manipolazione dei bit propria di un linguaggio per sistemi operativi quale è il C. Questo linguaggio è attualmente disponibile per C64, Apple 2 e PC IBM.

Nella versione per Commodore 64, il pacchetto contiene un vero e proprio sistema di sviluppo, comprendente: un editor a pieno schermo, potente quando un W.P., che consente di scrivere il codice sorgente con la massima facilità e comfort, un compilatore in passata singola, estremamente veloce (la versione per IBM compila circa 2000 linee al minuto), un vero e proprio sistema operativo con redirectione dell'output, comandi diretti per la gestione del disco (rename, copy, delete, display ecc.) definizione dei tasti funzione, presenza contemporanea di più programmi in memoria, eccetera.

La velocità di compilazione non è il solo vantaggio: anche la velocità di esecuzione non ha pari tra tutti i linguaggi implementati sul C64. Inoltre il codice oggetto prodotto dalla compilazione è tra i più compatti che si possano ottenere: la tabella riportata più sotto fornisce un confronto dettagliato con gli altri prodotti in circolazione.

La struttura del Promal rende superfluo il famoso (e famigerato) GOTO del Basic, poiché i numeri di linea non sono richiesti nemmeno per le funzioni di editor (come invece accade per i vari Pascal e macroassembler). Le strutture di controllo sono del tutto analoghe a quelle del Pascal, if then else, while, repeat, for e choose che funzionano esattamente come case, ma con in più l'opzione else; tuttavia, a differenza del Pascal, in Promal la stessa indentazione costituisce una direttiva per il compilatore. Al contrario i tipi di dati supportati si avvicinano di più allo standard del C: byte, word, integer e real. Concludiamo con una nota che dovrebbe dare l'idea della potenza e della professionalità di questo pacchetto: l'editor del Promal (12 K) è stato scritto interamente proprio in Promal. Il Promal costa 99.95 dollari ed è prodotto e distribuito da SMA (System Management Associates), Inc. 3325, Executive drive, Dept. C-1 Raleigh, North Carolina 27609 (USA).

Principali linguaggi disponibili per C64: comparazione delle prestazioni.

Linguaggio	esecuzione (secondi)	dimensione (bytes)	compilazione (secondi)
Basic interpretato	630	255	0
Comal	490	329	0
Forth	51	181	3.9
Pascal Abacus	55	415	108
SMA Promal	30	128	8.5

Vendo, Compro, Cerco, Scambio...

- **VENDO** Vic 20 con espansione 8K Ram a L. 150.000; eventuale monitor 7" a L. 120.000. Luciano Usuelli, Via Isonzo 31 - Varese. Tel. 0532/242596.
- **VENDO** super espansione 32K per ZX 80 (80K Rom) e ZX 81 con alimentatore con reset e cavo TV + libro "66 programmi per ZX 81" + cassetta originale 16K. L. 200.000 tutto. Alessandro Pini, Via Acquaregna 102 - Tivoli (RM). Tel. 0774/25649 (8.30/9.30 e 15.30/16.30).
- **CERCO** stampante e drive per Commodore 64. Paolo Piacentini, Via Cantine 18 - Rovato (BS). Tel. 030/722191 ore serali.
- **VENDO** grande cassa componenti + altoparlante 30 Watt, tutto a sole L. 65.000. Amplificatore 50 Watt quasi nuovo L. 310.000. Altoparlante 30 Watt e componenti L. 45.000. Renato Piccolo, Via Nicola Fabrizi 215 - Pescara.
- **VENDO** ZX Spectrum Plus + int. 1 + microdrive, tutto nuovo a L. 660.000. Vendo tavola grafica "RD Digitracer" per ZX Spectrum a L. 160.000. Giovanni Olivieri, Via Carducci 20, Mappano (TO). Tel. 011/9969046 ore serali.
- **VENDO** misuratore di campo della Prestel MC 20 mai usato a L. 250.000. Fernando Muratta, Viale Puglia 49 - A-ra-deo (LE).
- **VENDO** oscilloscopio Hameg 15 MHz monotraccia riv. N.E. frequenzimetro over-matic 350 MHz. Quadruplicatore di traccia per oscilloscopio. Voltmetro digitale LX 100. Tutto perfettamente funzionante a L. 700.000. Gianni Del Buono, Via Bellandi 7/A - Vado Ligure (SV). Tel. 019/888261.
- **VENDO** Spectrum 48K con tast. SAGA 1, interf. 1 + microdr. interf. tripla per joystick con ing. ROM, Alphacom 32 con rotoli carta, penna ottica, box sonoro e alimentatore, libri e pubblicazioni su Spectrum a L. 600.000. 10 cartucce piene di programmi + pacco cassette con oltre 500 giochi a L. 200.000. Claudio Soldi - Ostia (Roma). Tel. 06/5604307.

- **VENDO** Apple IIc usato pochissimo completo di monitor Apple 12 pollici, disk drive aggiuntivo, borsa originale, due alimentatori, mouse e decine di programmi originali a L. 2.400.000. stampante Image Writer 5 ore di lavoro effettivo a 1 milione. Se acquistati insieme, vendo a 3 milioni. Laura Passamonti, Via Tadino 42 - Milano. Tel. 02/2718768 (ore serali).
- **VENDO** Vic 20 (tre mesi di vita) con 20 giochi da favola, 2 cartridge (Poker, Space Battle), manuale, alimentatore, molte utilities, listati fantastici e una cassetta in omaggio a L. 150.000. Telefonare dopo le ore 17 a Paolo Uccino: 0942/792118.
- **VENDO** PC IBM compatibile 100%, 640K, 2 drive 360K, scheda multifunzione, monitor colore. L. 2.750.000. Programmi gestionali, contabilità, integrati, database. Roberto Rossi, Via Lario 26 - 20159 Milano
- **VENDO** Commodore 128 + registratore 1530 + monitor 1702 + joystick Quick Shot II. Tutto ancora imballato. Christopher Pecchia, Via Coroni 36 - Frosinone. Tel. 0775/854729 ore pasti.
- **VENDO** sintonizzatore TV Philips che trasforma un qualsiasi monitor in TV. Prezzo da concordare. Alessio Ricci, Via G. Bartolena 14 - Livorno. Tel. 0586/500901 ore 20.
- **VENDO** computer Sharp MZ-80K con video in bianco e nero e registratore incorporato. L. 400.000. Claudio Ballotta, Via F. Cavallotti 20 - Rovigo. Tel. 0425/29693 ore pasti.
- **VENDO** computer M.S.X. Sony 64K con circa 100 giochi, molte riviste e un libro sul Basic. Tutto a L.350.000. Giacomo Bergaglio, C.so S. Maurizio 35 - Torino. Tel.011/871043.
- **VENDO** per C64 programma "Tot professional" originale Commodore a prezzo stracciato. Giorgio Momeiti, Via Sebina 44 - Provaglio d'Isco (BS). Tel. 030/983897.
- **VENDO** Atari computer 800XL come nuovo + registratore + 2 libri istruzioni e 1 cartuccia Jungle Hunt + 9 cassette giochi + 2 cassette istruzioni Basic Atari con manuale a sole L.300.000 trattabili. Antonio Licciardello, Via E. Ferrario 31 - Gallarate (VA). Tel.0331/792825 Ore pasti.
- **VENDO** Spectrum Plus + int. Joystick + 400 programmi a L. 400.000. Giuseppe Fasulo, Via F. Berni 5 - Roma. Tel.06/7577351.
- **VENDO** oscilloscopio monotraccia Unahom G50D, ottimo stato a L.300.000 o permutato con stampante Commodore. Maurizio Violi, Via Molinetto di Lor.15 - Corsico (MI). Tel. 02/4407292.
- **VENDO** Commodore 128 + drive 1541 + stampante MPS 802 + Reg.C2N a L.1.200.000. All'acquirente regalo tutto il software di cui dispongo, imballi originali e garanzie intatte. Massimo Pucciarelli, V.le XX Settembre 132 - 54033 Carrara (MS).
- **VENDO** enciclopedia del Far da sé, Ed. Curcio, 96 fascicoli, 8 volumi rilegati al prezzo di L.250.000 trattabili. Giuseppe Maniglio, Via Kennedy 24 - Soletto (LE). Tel. 0836/667819 dopo le ore 19.
- **VENDO** computer Philips G.7000 seminuovo con istruzioni, alimentatore e 2 joystick + bellissimi giochi. A L.150.000 (valore L. 550.000). Manuel Tuozzi, Via G. Giro 12 - 45100 Rovigo.
- **VENDO** computer portatile Olivetti M10 adatto per giornalisti (24K) L. 1.200.000; Microplotter PL10 Olivetti L.300.000; programmi per M10: Multiplan su Eprom 32K L.240.000; Database L.60.000; Text-Ed L.60.000 su cassetta; calcolatrice Sharp EL7050 L.210.000; macchina da scrivere elettronica Olivetti Praxis 45D con due margherite a L. 1.100.000. Raffaele Savini, Largo Giannone 4 int.5 - Margherita di Savoia (FG). Tel.0883/754522.
- **VENDO** per C64/128 Magic Desk su disco e cassetta. Trasforma il computer in una perfetta macchina da scrivere. Vendo a L. 20.000. Ciro Gasparre, Via Cupa Camaldoli 12/A - 80131 Napoli.

Ritagliare e spedire in busta chiusa a:
Annunci di RadioELETTRONICA
 20122 Milano - Corso Monforte 39



Cognome

Nome

Via

Città

Testo dell'annuncio

.....

Sono abbonato

Sì

No

Verranno pubblicati solo gli annunci scritti in stampatello o a macchina.

PROGRAMMI HARDWARE E ACCESSORI



Stai per acquistare un personal computer Apple?

Vorrai sapere che cosa puoi farci. Questo volume è un aiuto indispensabile, una guida ragionata e completa di tutti i programmi, l'hardware e gli accessori disponibili in Italia. Prima di fare il tuo acquisto non puoi non consultarlo, perché solo se avrai a disposizione il programma, o i programmi, e le periferiche che li interessano, la tua scelta non ti lascerà deluso.

Hai appena acquistato un personal computer Apple?

Senza questa raccolta di programmi e le numerosissime segnalazioni di periferiche hardware e di accessori non potrai mai sapere quali e quanti utilizzi potrai farne.

Possiedi già da tempo un personal computer Apple?

Allora non c'è bisogno di dirti quanto può essere prezioso questo libro: sai già che il tuo computer, senza programmi e senza un hardware adeguato, è come un'auto senza benzina e senza le ruote. E poiché non c'è limite alla fantasia e all'inventiva, consultando questa guida scoprirai utilizzi impensati per il tuo personal. Utilizzi che ti permetteranno nel lavoro, nel tempo libero, nel gioco o nello studio, di essere sempre il più aggiornato, il più organizzato, il più soddisfatto...



In questa nuovissima edizione delle Pagine del Software per Apple trovi elencati, con una approfondita descrizione, tutti i programmi e le periferiche hardware disponibili in Italia.

**Nuova
edizione
aggiornata
e ampliata**

Le Pagine del Software e dell'Hardware sono un supplemento di **Applicando**, il mensile per i personal computer Apple. Acquistarle singolarmente costa 18.000 lire. Per chi si abbona ad **Applicando** sono in regalo.

Desidero ricevere, senza aggravio di spese postali, *Le Pagine del Software* con la seguente formula (segnare la casella di proprio interesse):

- 18.000 lire per ricevere l'edizione 1985-86 di **Le pagine del Software e dell'Hardware per Apple.**
- 60.000 lire per ricevere 10 numeri di **Applicando** e in regalo l'edizione 1985-86 di **Le pagine del Software e dell'Hardware per Apple.**

COGNOME E NOME

VIA N.

CAP. CITTÀ PROV.

Allego assegno non trasferibile di L. inestato a Editronica Srl.

Allego ricevuta di versamento di L. sul CC postale N. 19740208 inestato a Editronica Srl.

Corso Monteforte 39, 20122 Milano.

Pago fin d'ora con la mia carta di credito Bankamericard N. scadenza

Data Firma

ritagliare, compilare e spedire a: Editronica Srl, Corso Monteforte 39, 20122 Milano.



MELCHIONI presenta in esclusiva il ricevitore scanner HANDIC 1600

16 canali programmabili su 4 bande: 68-88 MHz, 138-174 MHz, 380-512 MHz, e la banda aeronautica 108-136 MHz. Canale prioritario, funzione di ricerca, possibilità di scansione entro una determinata banda. Funzioni di lock-out e di ritardo sulla tastiera. Sensibilità elevata su tutte le frequenze. Il piccolissimo scanner Handic 1600 (60 x 160 x 180 mm) e dotato di vox interno e di pile per il back-up della memoria. Viene fornito completo di staffa per il montaggio automobilistico e di cordone di alimentazione a 12 volt.



INTERNO

handic

MELCHIONI ELETTRONICA

20135 Milano - Via Friuli, 15-18 - tel. 57941 - Filiali, agenzie e punti di vendita in tutta Italia
Centro assistenza: DE LUCA (12 DLA) - Via Astura, 4 - Milano - tel. 5696797